

# Virtualization of CubeSat Downlink Ground Stations Using the APRS I-Gate Network

Greg Linton

A Thesis  
Submitted to the University of Manitoba  
In partial fulfillment of  
the requirements of the degree of  
Master of Science  
in  
the Department of Electrical and Computer  
Engineering  
University of Manitoba  
Winnipeg, Manitoba, Canada



UNIVERSITY  
OF MANITOBA

September 27, 2016



Creative Commons Attribution 4.0 International License





# Virtualization of CubeSat Downlink Ground Stations Using the APRS I-Gate Network

Greg Linton

A Thesis

Submitted to the University of Manitoba

In partial fulfillment of

the requirements of the degree of

Master of Science

in

the Department of Electrical and Computer

Engineering

University of Manitoba

Winnipeg, Manitoba, Canada



UNIVERSITY  
OF MANITOBA

September 27, 2016

Front Matter {28} + Main Matter {211} = 239



Creative Commons Attribution 4.0 International License

---

“Our two greatest problems are gravity and paper work.  
We can lick gravity,  
but sometimes the paperwork is overwhelming.”  
~ Wernher von Braun [Diaz2016]

# Abstract

This thesis presents the design, simulation and analysis of a new network based on the Automatic Packet Reporting System (APRS) concept to enhance communications of CubeSat. Created in the 1980s, APRS is arguably one of the world's largest Radio Frequency (RF) networks maintained by a community of licensed amateur radio volunteers for a variety of terrestrial purposes. Using these established nodes we can create a virtualized ground station network for satellite downlinks.

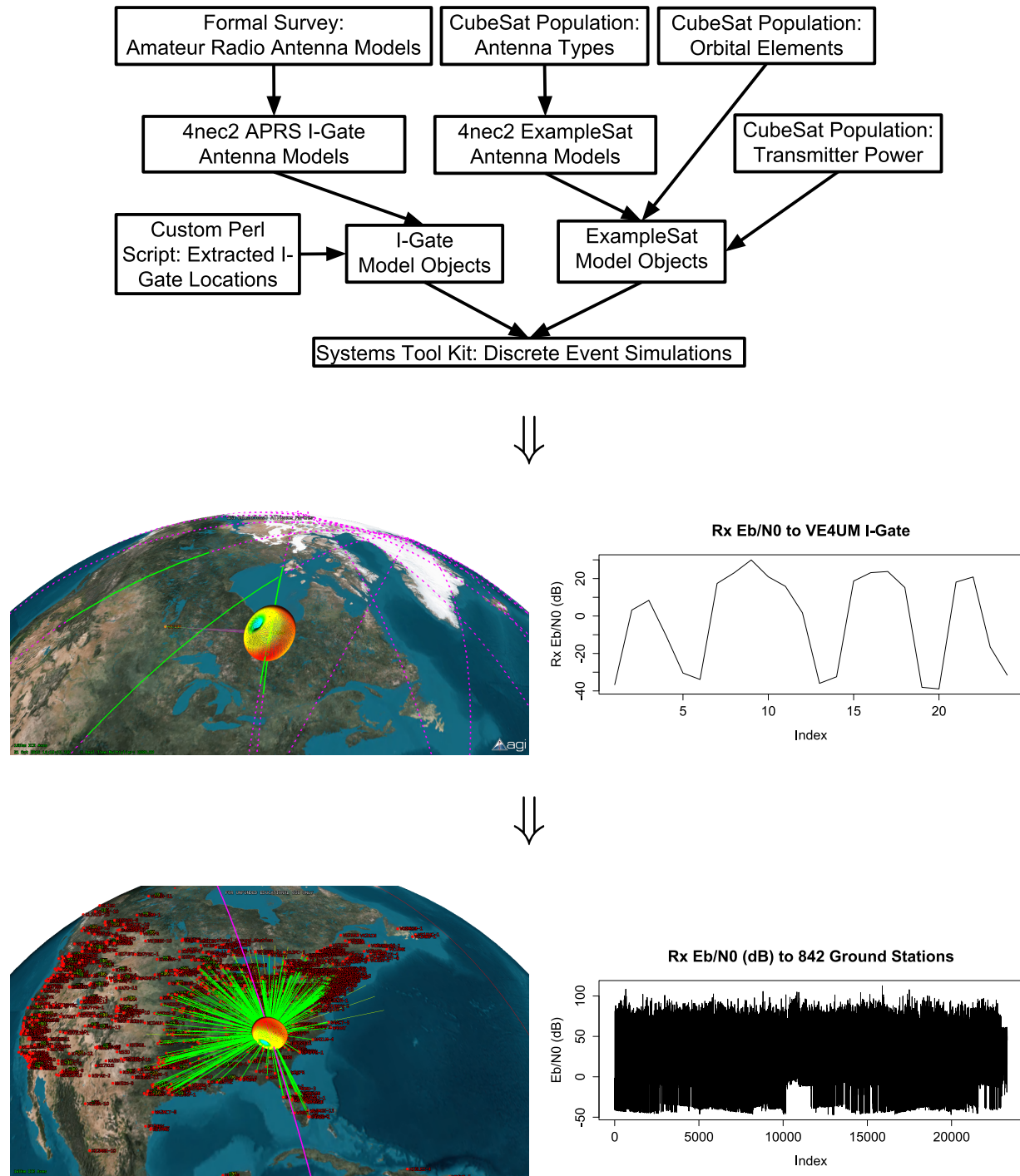
Based on literature reviews of CubeSat communications systems and their orbital elements, as well as direct surveys of the amateur radio community, comprehensive far-field antenna models were created in 4nec2 for both the ground and space sections of the CubeSat downlink. The ground station antenna models include: a directional AMSAT/OSCAR Yagi-Uda, an omnidirectional J-pole, ground plane, and a whip antenna. The CubeSat antenna models used include dipoles, monopoles, and turnstiles. These models were evaluated and then imported into the Analytical Graphics Inc (AGI) Systems Tool Kit (STK) to form a discrete event simulation based on nonhomogeneous WGS84 location data extracted from real world APRS Internet Gateway (I-Gate) packets. All objects within the simulation use realistic antenna models, node locations, propagation models and satellite orbital mechanics. This simulation can act as a suite of generally applicable example satellites models (ExampleSat) for analytical comparisons that include link budgets, access times, differing data rates, antenna designs, orbital altitudes and ground station locations. Furthermore, it represents a North American continental scale RF satellite communications downlink network model.

Using the virtual ground station network alters the net probability distribution of the received signal from the spacecraft. Specifically, the thesis compares non-stationary time series analysis methods upon the net received signal. The virtualized ground station network enables the aggregated received signal to appear stationary. This in turn may allow for alternative signals analysis techniques to improve a CubeSat's downlink performance. The simulation allows us to generate representative received signals from ExampleSat to the ground station nodes for exploratory analysis, utilizing less arbitrary assumptions.

This simulation and the set of models enable a more rapid start to trade-off studies for link budget design. This set of ExampleSat downlink models can be used by new designers of small-satellite communications systems.



# Visual Abstract





# Acknowledgments

In no particular order:

## Research and Funding Support

- Thank you to National Defence and all the staff at the Canadian Forces School of Aerospace Studies for enabling me to complete my studies. Without my employer's exceptional commitment to life-long learning, I would not be here.

## Technical Support

- Thank you to the developer of FindU.com, Steve Dimse (K4HG) who answered all my neophyte questions about the subtleties of APRS and his exemplary software service.
- Thank you to everyone behind APRS.fi for bringing the capability of a Common Operating Picture to the APRS community without the users needing specialized software.
- Thank you to my brother Corrie (VE4COR) for helping me to set up equipment and experiment with software defined radios. And, for generally being awesome.
- Thank you to my friend Rob (Not a Ham yet) for programming assistance and general expertise. Your support, participation and desire to do “more interesting projects” got the ball rolling.
- Thank you to my friend and University of Manitoba alumni Matt (VE4MDW), for programming assistance and grad studies advice. I look forward to more fun projects in the future.
- Thank you to my friend and University of Manitoba alumni Dario (VE4SCH), for being an example of how to do graduate studies. For your wonderful help with editing. You're an inspiration for everyone you meet.
- Thank you to Amy Dario, Graduate Student Advisor with the University of Manitoba Department of Electrical and Computer Engineering. Without your dedicated support this wouldn't have finished.

## Audio Support

Thanks overall to The Crystal Method, The Chemical Brothers, The Art of Noise, The Heavy, Nine Inch Nails, Fatboy Slim, Propellerheads, John Lee Hooker, Black Grape / Happy

---

Mondays, The Black Keys, Blue Man Group, Buck 65, Butthole Surfers, Breakbeat Era, Ben Folds Five, deadmau5, Creedence Clearwater Revival, The Charlatans, Cake, Gorillaz, Pop Will Eat Itself, Massive Attack, Primus, Portishead, Underworld, Garbage, Glitch Mob, Weird Al Yankovic, Danko Jones, Aphex Twin, Caravan Palace, and all the others I should have listed. Thank you to online music services such as Digitally Imported, Google Music, SoundCloud, CBC Radio, Creative Commons Mixter, and all the others without whom the long hours would have been eternal. A special thank you to Daft Punk for the Tron: Legacy Soundtrack, I consider it the hidden layer to this thesis and all my graduate studies leading up to it.

## Personal Support

- Thank you to my thesis adviser, Dr. Witold Kinsner (VE4WK), for somehow convincing and encouraging me to do all this, for backing me when I needed help.
- Thank you to my father Woody (VE3JJA) for putting the idea for the thesis into my head. Its a question that needed to be answered. No one else could tell me the answer, so I had to go figure it out.
- Thank you to my friend Myrna (Not a Ham yet) for her ongoing questions of “how’s that thesis going”. Motivation = Momentum. For your wonderful help with editing.
- Thank you to my friend Liz (Not a Ham yet), for telling me I’d be capable of doing all this. Can I blame you for all of this?
- Thank you to my friend Ahmad (VE4SAT), for your enthusiasm and encouragement.
- Thank you to all the team members past, present and future at UMSATS. You all have my respect and admiration for joining me in the shared dream of putting a tiny box into space. Or other crazy projects, like surviving a 26 hour drive...twice in one week. And doing it all for no pay, no credits. Only glory.
- Thank you Erin (Not a Ham yet) for your support, understanding and patience. This entire part-time academic while full-time employed gig has eaten up time that rightly should have been spent at your side. You seem to understand my deep seated need for geekyness, and this effort is finally coming to fruition.

## Thank You

Thank you for joining me on this meandering journey through academia and technology. Together we’ve explored old radio formats and proposed new ones. We have peered at places firmly on the Earth from places abstractly in space.



# Table of Contents

<b>Abstract</b>	<b>i</b>
<b>Visual Abstract</b>	<b>iii</b>
<b>Acknowledgments</b>	<b>v</b>
<b>Nomenclature</b>	<b>7</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Nanosatellites and CubeSats . . . . .	1
1.2 Problem Statement . . . . .	2
1.2.1 Motivation . . . . .	2
1.2.2 Problem Definition . . . . .	3
1.2.3 Proposed Solution . . . . .	4
1.3 Thesis Formulation . . . . .	4
1.3.1 Aim of the Thesis . . . . .	4
1.3.2 Core Concepts within the Thesis . . . . .	4
1.3.3 Thesis Statements . . . . .	4
1.3.4 Scope of the Thesis . . . . .	5
1.3.5 Research Questions . . . . .	5
1.3.6 Thesis Organization . . . . .	7
<b>2 Background and Literature Review</b>	<b>9</b>
2.1 Satellite Communications Networks . . . . .	9
2.1.1 Example LEO Satellite Communications Downlink Networks . . . . .	11
2.2 ExampleSat-165 . . . . .	17
2.3 Virtualization and Overlay Networks . . . . .	17
2.3.1 Network Virtualization . . . . .	17
2.3.2 Overlay Network . . . . .	18
2.4 Automatic Packet Relay System . . . . .	18
2.4.1 Space vs Terrestrial Communications . . . . .	21
2.4.2 Audio Frequency Shift Keying (AFSK) Modulation . . . . .	22
2.4.3 AX.25 UI Packet Format . . . . .	26
2.4.4 APRS-IS Network Topology . . . . .	28
2.4.5 I-Gate Hardware Design . . . . .	29
2.4.6 APRS-IS Software/Firmware Design . . . . .	31
2.4.7 APRS “q-construct” . . . . .	32

2.5	CubeSat Population Survey . . . . .	33
2.5.1	Orbital Design Impacting Telecommunications Design . . . . .	34
2.5.2	CubeSat Orbits . . . . .	37
2.5.3	CubeSat Downlink Selection . . . . .	41
2.5.4	CubeSat Downlink Modulation, Baud & Protocol . . . . .	45
2.5.5	CubeSat Antenna Designs . . . . .	47
2.6	Metrics of Performance . . . . .	48
2.6.1	Bit Error Rate and $E_b/N_0$ . . . . .	48
2.6.2	Stationarity of $E_b/N_0$ . . . . .	53
2.7	Doppler Shift . . . . .	60
2.7.1	Doppler induced bit errors within I-Gates . . . . .	62
2.8	Summary . . . . .	66
<b>3</b>	<b>System Design &amp; Development</b>	<b>67</b>
3.1	Model & Simulation Design . . . . .	67
3.1.1	Model Composability . . . . .	68
3.1.2	Amateur Radio Antenna Survey . . . . .	69
3.1.3	I-Gate Locations . . . . .	72
3.2	Modelling Tool Selection & Implementation . . . . .	73
3.2.1	HFSS . . . . .	73
3.2.2	FEKO . . . . .	74
3.2.3	4nec2 . . . . .	74
3.2.4	Systems Tool Kit . . . . .	76
3.3	R programming language . . . . .	80
3.4	I-Gate Emulation . . . . .	80
3.5	Summary . . . . .	84
<b>4</b>	<b>System Implementation</b>	<b>85</b>
4.1	Scenario Overview . . . . .	85
4.2	I-Gate Location extraction . . . . .	86
4.3	STK Simulations . . . . .	89
4.3.1	STK Scenario Setup . . . . .	89
4.3.2	STK Objects . . . . .	90
4.3.3	Modulation within STK . . . . .	93
4.3.4	STK CommSystem . . . . .	93
4.4	Antenna Models . . . . .	94
4.4.1	CubeSat Antenna Models . . . . .	95
4.4.2	I-Gate Ground Station Antennas . . . . .	97
4.4.3	STK Variables and Observables . . . . .	103
4.5	Verification and Validation . . . . .	105
4.5.1	Model & Simulation Verification . . . . .	105
4.5.2	Model & Simulation Validation . . . . .	105
4.6	Summary . . . . .	107

<b>5</b>	<b>Experimental Results and Discussion</b>	<b>109</b>
5.1	Overview . . . . .	109
5.2	STK Outputs . . . . .	109
5.3	Packet Reception . . . . .	113
5.4	Non-Stationary Time Series Analysis . . . . .	117
5.4.1	Slope Comparison . . . . .	121
5.4.2	ANOVA . . . . .	129
5.4.3	Normality Testing . . . . .	130
5.5	Phillips-Perron Stationarity Testing . . . . .	135
5.5.1	Phillips-Perron p-value ratios . . . . .	138
5.6	Doppler Shift Results . . . . .	142
5.7	Summary . . . . .	145
<b>6</b>	<b>Thesis Conclusions</b>	<b>147</b>
6.1	Review . . . . .	147
6.1.1	Considerations . . . . .	148
6.2	Contributions . . . . .	148
6.3	Extensions To The Work . . . . .	150
6.3.1	Amateur Radio Community Surveys . . . . .	151
6.3.2	I-Gates and Cognitive Radio . . . . .	151
6.4	Limitations and Potential Solutions . . . . .	155
6.5	Personal . . . . .	156
	<b>References</b>	<b>159</b>
<b>A</b>	<b>Appendix: Modelling</b>	<b>169</b>
A.1	Ground Station Antenna models . . . . .	169
A.2	Orbital Mechanics . . . . .	173
A.3	Parameters Required to Analyze the Downlinks . . . . .	173
A.3.1	Losses . . . . .	173
A.3.2	Satellite Transmission system parameters . . . . .	178
A.3.3	Ground Receiver system parameters . . . . .	179
A.3.4	Criteria of the Digital Reception . . . . .	181
<b>B</b>	<b>Appendix: Surveys</b>	<b>183</b>
<b>C</b>	<b>Appendix: Source Code</b>	<b>189</b>
C.1	APRS I-Gate Call Signs and Locations . . . . .	189
C.1.1	record . . . . .	189
C.1.2	sort . . . . .	192
C.1.3	getinfo . . . . .	193
C.1.4	R statistical programming sample . . . . .	195
C.2	Tools Used . . . . .	199

<b>D</b>	<b>Appendix: Supplementary Analysis Data, Results and Methods</b>	<b>203</b>
D.1	Time Series Analysis . . . . .	204
D.1.1	Short Time Window Analysis of Non-Stationary Time Series . . . . .	204
D.1.2	Shapiro-Wilks . . . . .	204

# List of Figures

2.1	STK scenario of CubeSat downlink. . . . .	10
2.2	Model scenario of a satellite at 165Km altitude, sun-synchronous orbit and the resulting LOS from the U of M. . . . .	11
2.3	NASA Near Earth Network is an example of multiple redundant ground stations [SCaN2014]. . . . .	12
2.4	Unconfirmed Global Educational Network for Satellite Operators (GENSO) Ground Station Servers (GSS). . . . .	13
2.5	FUNcube. Photo from Wikimedia Commons (Photo taken by Wouter Wegge- laar). . . . .	14
2.6	A FUNcube Pro USB Dongle. From: Wikimedia Commons (Photo by Jeff Keyzer). . . . .	15
2.7	Example AMSAT/OSCAR Yagi-Uda antenna ground station antenna. From: Wikimedia Commons SwissCube Ground Station (Photo by Olivier Chatelain). . . . .	16
2.8	Common Operating Picture enabled through APRS connected nodes. From: APRS.fi . . . . .	19
2.9	APRS node “heat map” of North America. Red indicates a higher density of nodes. From: APRS.fi . . . . .	20
2.10	Physical diagram of ExampleSat communications downlink. . . . .	22
2.11	Physical & logical diagram of the APRS Network. . . . .	23
2.12	AFSK audio baseband modulation frequency distribution. . . . .	23
2.13	Diagram of the APRS-IS network. After: Geoff Gatward, VK2XJG [Gatward2009]. . . . .	28
2.14	Different I-Gate station configurations. . . . .	30
2.15	Example APRS I-Gate ground station antenna models in 4nec2. . . . .	31
2.16	Downlink with associated losses. After:[King2008]. . . . .	34
2.17	Example TLE. After: NASA [NASA2011] . . . . .	36
2.18	Eccentricity as it impacts ExampleSat’s slant range. . . . .	37
2.19	Histogram of CubeSat Orbit Eccentricity. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	38
2.20	Histogram of CubeSat Orbit Inclinations. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	39
2.21	Histogram of CubeSat Orbit Altitude. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	40
2.22	ExampleSat eclipse zone with Earth’s day / night terminator. . . . .	41
2.23	ExampleSat-165’s orbit based on CubeSat community survey data. . . . .	42
2.24	Histogram of CubeSat Downlink Transmitter Power. Data derived from Klo- fas <i>et al.</i> [Klofas2013]. . . . .	43
2.25	Histogram of CubeSat Downlink Frequencies in VHF and UHF. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	44

2.26	Histogram of CubeSat Downlink Frequencies in UHF only. After:[Klofas2013]	45
2.27	Example Dipole antennas . . . . .	48
2.28	Possible models of ExampleSat-156's antenna design in 4nec2. . . . .	49
2.29	Changes in slant range induce non-stationarity of revived $E_b/N_0$ over time. .	54
2.30	White and Brown Noise time series data generated via power law. . . . .	55
2.31	Plots of White Noise moments and Autocovariance. . . . .	56
2.32	Plots of Brown Noise moments and Autocovariance. . . . .	57
2.33	Non-stationarity of the received $E_b/N_0$ for ExampleSat to a single ground station. Note that Index is unitless. . . . .	58
2.34	Doppler shift impact vs APRS channel notch filter. . . . .	64
2.35	Potential Doppler shift could induce errors into the bit stream. . . . .	65
3.1	Logical diagram of the model and simulation set. . . . .	69
3.2	APRS node "heat map" of North America. From: APRS.fi . . . . .	73
3.3	APRS I-Gate node map for much of North America. . . . .	74
3.4	Example ground station antenna models in 4nec2. . . . .	75
3.5	ExampleSat-165 antenna models rendered in 4nec2. . . . .	76
3.6	ExampleSat-165 with a "turnstile" far field antenna pattern in orbit above North American APRS I-Gate locations. . . . .	77
3.7	Far field patterns of ExampleSat-165's and a single I-Gate. . . . .	78
3.8	Object browser view. . . . .	79
3.9	GNU Radio graphical interface. . . . .	81
3.10	Local network physical diagram with SDR Satellite Channel Emulator and APRS I-Gates. . . . .	83
4.1	STK Scenario with multiple ExampleSats, and I-Gate locations. . . . .	86
4.2	APRS I-Gate locations across North America imported via KML into STK simulation software. . . . .	87
4.3	Plot of Percent Uniqueness of I-Gates vs Time. . . . .	88
4.4	Differences in ThetaPhi coordinate systems between modeling and simulation tools. . . . .	94
4.5	STK ThetaPhi coordinate system. From: [AGI2016a] . . . . .	95
4.6	ExampleSat-165's VHF dipole antenna design in 4nec2. . . . .	97
4.7	VE4UM directional station antenna model in 4nec2. . . . .	100
4.8	GP144-435 omnidirectional antenna model in 4nec2. . . . .	101
4.9	J-Ant144 omnidirectional antenna model in 4nec2. . . . .	102
4.10	2mGP omnidirectional antenna model in 4nec2. . . . .	103
4.11	Diagram of possible variables and observables from STK . . . . .	104
5.1	Received $E_b/N_0$ from ExampleSat-165 to VE4UM AMSAT/OSCAR Yagi-Uda antenna (60 s interval). . . . .	111
5.2	Received $E_b/N_0$ from ExampleSat-165 to VE4UM AMSAT/OSCAR Yagi-Uda antenna (1 s interval). . . . .	111
5.3	Received $E_b/N_0$ from ExampleSat-165 to VE4UM I-Gate using 2mGP (60 s interval). . . . .	112

5.4	Received $E_b/N_0$ from ExampleSat-165 to VE4UM I-Gate using 2mGP (1 s interval).	112
5.5	Received $E_b/N_0$ from ExampleSat to all North American I-Gates.	113
5.6	The first 500 samples from Fig. 5.5.	114
5.7	Plot of Probability of Packet reception vs Received $E_b/N_0$ .	115
5.8	Plot of Probability of Packet reception vs Received BER.	116
5.9	Plot of Received BER vs Received $E_b/N_0$ with residuals.	117
5.10	Science Payload Data Probability of Packet Reception vs BER.	118
5.11	Bulk Science Payload Data Probability of Packet Reception vs BER.	119
5.12	White and Brown Noise time series data generated via power law.	120
5.13	Plot of White Noise short time window means vs window number with comparative slopes; window sizes 50 and 700 samples.	121
5.14	Plot of White Noise short time window means vs window number with comparative slopes.	122
5.15	Plot of White and Brown Noise short time window means vs window number, axis adjusted for simple visual comparisons.	124
5.16	Plot of $E_b/N_0$ Window mean value vs window number with comparative slopes for all short time window sizes.	125
5.17	Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.	126
5.18	Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.	127
5.19	Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.	128
5.20	Plot showing all three time series Linear Best Fit lines for comparison.	128
5.21	Normal QQ Plot of Rx $E_b/N_0$ raw data series.	132
5.22	Normal QQ Plots of Rx $E_b/N_0$ from 0 to 50 and 51 to 100.	133
5.23	White Noise Shapiro-Wilk Normality Test p-value vs Window Size.	134
5.24	Brown Noise Shapiro-Wilk Normality Test p-value vs Window Size.	134
5.25	$E_b/N_0$ Time Series Shapiro-Wilk Normality Test p-value vs Window Size.	135
5.26	Non-stationarity of the received $E_b/N_0$ for ExampleSat-165 to the VE4UM I-Gate.	137
5.27	White Noise Phillips-Perron Stationarity Test p-value vs Window Size.	139
5.28	Brown Noise Phillips-Perron Stationarity Test p-value vs Window Size.	140
5.29	Received $E_b/N_0$ time series Phillips-Perron Stationarity Test p-value vs Window Size.	140
5.30	Ratio of Phillips-Perron Stationarity Test p-value vs Window Size for all three time series.	141
5.31	Doppler Shift and Doppler Rate seen by VE4UM's single I-Gate from ExampleSat-165.	142
5.32	The impact of Doppler shift on the demodulated AX.25 audio baseband signal.	143
6.1	Cognitive Radio Problem Definition. After [Fortetsanakakis2011]	152
A.1	VE4UM directional station antenna model in 4Nec2.	170

A.2	GP144-435 omnidirectional antenna model in 4Nec2. . . . .	171
A.3	J-Ant144 omnidirectional antenna model in 4Nec2. . . . .	171
A.4	2mGP omnidirectional antenna model in 4Nec2. . . . .	172
D.1	Plot of White Noise short time window means vs window number with comparative slopes; window sizes 50 and 700 samples . . . . .	204
D.2	Plot of White Noise short time window means vs window number with comparative slopes . . . . .	205
D.3	Plot of Brown Noise short time window means vs window number with comparative slopes . . . . .	205
D.4	Plot of White Noise short time window means vs window number, axis adjusted for comparisons . . . . .	206
D.5	Plot of Brown Noise short time window means vs window number, display adjusted for comparisons . . . . .	206
D.6	Plot of Brown Noise short time window means vs window number, display adjusted for comparisons . . . . .	207
D.7	Plot of $E_b/N_0$ Window mean value vs window number with comparative slopes for all short time window sizes . . . . .	208
D.8	White Noise Shapiro-Wilk Normality Test p-value vs Window Size . . . . .	209
D.9	Brown Noise Shapiro-Wilk Normality Test p-value vs Window Size . . . . .	210
D.10	$E_b/N_0$ Time Series Shapiro-Wilk Normality Test p-value vs Window Size . . . . .	211



# List of Tables

2.1	Access times for the example satellite shown in Fig. 2.2. . . . .	11
2.2	AX.25 UI Packet Format in bytes for terrestrial APRS. After: [Lustig2015] .	26
2.3	AX.25 UI Packet Format in bytes for High Altitude / Space. After: [Lustig2015]	27
2.4	Potential ExampleSat altitudes (suffixes) based on Fig. 2.21. . . . .	40
2.5	Satellite Downlink RF Bands in Canada[Shtun1995]. . . . .	42
2.6	Amateur Satcom bands in Canada [Shtun1995]. . . . .	43
2.7	Distribution of UHF downlink Satcom frequencies. After:[Klofas2013] . . . .	44
2.8	The population of protocol types within CubeSats. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	46
2.9	The population of modulation types within CubeSats. Data derived from Klofas <i>et al.</i> [Klofas2013]. . . . .	46
2.10	Population of CubeSat data bauds. After: [Klofas2013] and [Buscher2014]	47
2.11	Population of CubeSat antenna designs. After: [Klofas2013] and [Buscher2014]	47
2.12	$r_{BER}$ to $E_b/N_0$ for APRS. Data derived from Wallio and Goode[Wallio2015, Goode1984]. . . . .	51
2.13	Single ground station simulation results of induced Doppler Rate. . . . .	63
3.1	Survey of I-Gate use. . . . .	70
3.2	Survey of AMSAT/OSCAR ground station use. . . . .	70
3.3	Survey of AMSAT/OSCAR ground station antennas. . . . .	70
3.4	Antennas models selected with input from the community survey. . . . .	72
4.1	Sample table of Uniqueness for APRS I-Gates. . . . .	88
4.2	Example PhiTheta far field pattern file. . . . .	96
4.3	CubeSat antenna models created in 4nec2 and imported into STK. . . . .	96
4.4	I-Gate antenna models created in 4nec2 and imported into STK. . . . .	97
5.1	Access Summary Report from ExampleSat-165 to VE4UM I-Gate. . . . .	110
5.2	Link Budget Report from ExampleSat-165 to VE4UM I-Gate (1 s interval). .	110
5.3	Access Times comparisons. . . . .	113
5.4	Science Data Payload to Total Packet Volume. . . . .	116
5.5	ANOVA: Single Factor: White Noise Window Size. . . . .	129
5.6	ANOVA: Single Factor: Brown Noise Window Size. . . . .	130
5.7	ANOVA: Single Factor: $E_b/N_0$ time series data Window Size. . . . .	130

5.8	ANOVA: Two Factor Without Replication: White Noise, Brown Noise & $E_b/N_0$ time series data vs Window Size. . . . .	131
5.9	Phillips-Perron Stationarity Testing. . . . .	138
5.10	Doppler shift and Doppler Rate received by VE4UM I-Gate from ExampleSat-165 over 24 hours. . . . .	143
A.1	Ionospheric Losses with frequency. After: [Thakker2010] . . . . .	176
A.2	Channelization of AMSAT frequency bands. After:[Shtun1995] . . . . .	181
B.1	Survey of I-Gate use . . . . .	183
B.2	Survey of I-Gate use 2 . . . . .	183
B.3	Survey of AMSAT/OSCAR ground station use . . . . .	183
B.4	Survey of AMSAT/OSCAR ground station antennas . . . . .	184
B.5	Survey of I-Gate antenna types . . . . .	184
B.6	Survey of AMSAT/OSCAR ground station use . . . . .	185
B.7	Community survey of mobile APRS beacon transmit power . . . . .	187
B.8	Community survey of mobile APRS beacon interval . . . . .	187
D.1	Access Summary Report from ExampleSat-165 to VE4UM I-Gate . . . . .	203
D.2	Link Budget Report from ExampleSat-165 to VE4UM I-Gate (1 s interval) . . . . .	203
D.3	White Noise Shapiro-Wilk Normality Test . . . . .	208
D.4	Brown Noise Shapiro-Wilk Normality Test . . . . .	209
D.5	Rx $E_b/N_0$ Shapiro-Wilk Normality Test . . . . .	210

# Nomenclature

## List of Abbreviations

ACF	Autocorrelation Function
ADF	Augmented Dickey Fuller test.
ADSL	Asymmetric Digital Subscriber Line
AFSK	Auido Frequency Shift Keying
AGI	Analytica Graphics Incorporated
ANOVA	Analysis of Variance: a suite of statistical models used to determine the source of variation among and/or between groups
API	Application Program Interface
APRS-IS	Automatic Packet Reporting System Internet-Service
APRS	Automatic Packet Reporting System
ARIMA	Autoregressive Integrated Moving Average
Bd	baud
BEM	Boundary Element Method
BER	Bit Error Rate
BFSK	Binary Frequency Shift Keying
BPSK	Binary Phase Shift Keying
CETS	Connectionless Emergency Traffic System
COTS	Commercial Off the Shelf; non-developmental items available within the common commercial marketplace. Typically they are considered consumer grade electronics and devices.
CR	Cognitive Radio
DF	Dickey Fuller test.
DTED	Digital Terrain Elevation Data

---

Eb	Energy Per Bit
ECI	Earth Centric Inertial reference frame
EIRP	Effective Isotropic Radiative Power [dBW]
ELFIN	Electron Losses and Fields Investigation
FDMA	Frequency Division Multiple Access
FEM	Finite Element Method
FSK	Frequency Shift Keying
FSPL	Free Space Path Loss
GENSO	Global Educational Network for Satellite Operators
GUI	Graphical User Interface
HFSS	High Frequency Structure Simulator
IDE	Integrated Development Environment
ISM	Industrial, Scientific and Medical radio frequency band
ITU	International Telecommunications Union
JFCC Space	Joint Functional Component Command for Space. A part of the US Strategic Command.
LEO	Low Earth Orbit
LNA	Low Noise Amplifier
LNA	Low Noise Amplifier
LOS	Line of Sight
MoM	Method of Moments
N0	Noise Power Spectral Density
NRZI	Non-Return to Zero Inverted
OSI model	Open Systems Interconnection model
PACF	Partial Autocorrelation Function
PEC	Perfect Electric Conductor
PER	Packet Error Rate

---

PP Test	Phillips-Perron stationarity test. It assumes a null-hypothesis that a unit root is present in the data series and thus the series is not stationary.
PSD	Power Spectral Density
QoS	Quality of Service
RAAN	Right Ascension of the Ascending Node
RHC	Right Hand Circular
SDMA	Space Division Multiple Access
SDR	Software Defined Radio
FWWSS	Finite Window Weak Sense Stationarity
SPM	Simplified Perturbations Models
SSID	Service Set Identifier; A network unique identifier for a radio systems across the APRS network
STK	Systems Tool Kit
TDMA	Time Division Multiple Access
TLE	Two Line Element set which describes the orbital trajetory of any space object. The format of the set is proscribed the the Joint Functional Component Command for Space with the US Strategic Command.
TNC	Terminal Node Controller
TTC	Tracking Telemetry and Command
TUB SSD	Technische Universität Berlin Small Satellite Database
TUB SSD	TUB Small Satellite Database
TUB	Technische Universität Berlin: Technical University of Berlin
TUB	Technische Universität Berlin
UI frame	Un-numbered Information frame
USRP	Universal Software Radio Peripheral
USSTRATCOM	United States Strategic Command
Wireless MAN	Wireless Metropolitan Area Network
w	BandWidth
APRS	Amateur Packet Relay System

ECF	Earth Centric Fixed reference frame
FOV	Field of View
NASA NEN	NASA Near Earth Network
RAAN	Right Ascension of the Ascending Node
SSN	Space Surveillance Network
STK	Systems Tool Kit

### List of Definitions

AMSAT	The collective name of amateur radio satellite organizations worldwide.
downlink	The radio frequency communications link from the satellite in orbit to the ground station terminal on the Earth's surface.
I-Gate	an APRS Internet Gateway that converts between RF packets to TCP/IP based traffic for posting on the Tier 2 network
model	A representation of a system, entity, pheomenon, or process. Software models of specific entities are comprised of algorithms and data. A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon or process.
q-construct	a set of conditional characters with a q added to APRS packet that enables the APRS-IS transport layer.
simulation	The execution over time of models representing the attributes of one or more entities or processes. Human-in-the-Loop simulations, also known as simulators, are a special class of simulations. A method of implementing a model over time.

### List of Mathematical Symbols

$\alpha_{point}$	the ground station pointing error [degrees]
$\delta$	minimum elevation angle [degrees]
$\lambda$	wavelength [m]
$\Omega$	the orbit's Right ascension of the ascending node (RAAN) [degrees]
$\omega$	the orbit's argument of perigee [degrees]
$\Phi$	antenna beamwidth [degrees]
$\tau$	the orbital period [minutes]
$a$	semi-major axis of the orbit [km]

---

$c$	speed of light in a vacuum
$c_{awgn}$	additive white Gaussian noise channel capacity
$C_{gndline}$	the ground station transmission line coefficient
$l_{conf}$	Confidence Level
CDMA	Code Division Multiple Access
$d$	distance in km
$dD/dt$	rate of change of Slant Range from the ground station to the satellite in its orbit.
$\Delta f$	deviation from the centre carrier frequency
$E_b$	Energy per Bit; this notation breaks convention of using non-bold capitals for set scripts only
$\epsilon_m$	maximum elevation angle possible for the ground station's antenna
$f_b$	The bit rate of the downlink signal
$f_c$	centre frequency
$f_i$	Transmit frequency of index $i$
$f_m$	the mark (1) frequency within Binary Frequency Shift Keying modulation
$f_s$	the space (0) frequency within Binary Frequency Shift Keying modulation
$f_d/f_c$	ratio of frequency change in Hz/sec relative to centre frequency of the satellite's transmitter
$G_{ant}$	Gain of the antenna system [dB]
$G_r$	gain of the receiver in [dB]
$G_t$	gain of the transmitter in [dB]
$GM$	universal gravitational constant times the mass of the Earth
$h$	mean altitude of the satellite above the Earth's surface (assuming an eccentricity of approximately zero) [km]
$h$	the orbit's mean height above the Earth's surface [km]
$h$	the satellite's mean height above the Earth's surface [km]
$h_{iono}$	mean height of the absorbing layer of the ionosphere [km]
$i$	orbital inclination of the satellite

---

---

$k$	Boltzmann's constant [dBW/K/Hz]
$h_p$	height of perigee [km]
$L_{atmos}$	attenuation due to the atmosphere [dB]
$L_{connect}$	losses due to connectors [dB]
$L_{gndline}$	losses due to the ground station transmission line [dB]
$L_{gndpoint}$	loss due to pointing from the ground station antenna [dB]
$L_{iono}$	attenuation loss due to the ionosphere [dB]
$L_{line}$	loss due to the transmission line [dB]
$L_{path}$	the simplified free space path loss [dB]
$L_{point}$	loss due to antenna pointing alignment [dB]
$L_{polar}$	loss due to antenna polarization misalignment [dB]
$L_{rain}$	attenuation due to rain [dB]
$L_S$	Free Space Path Loss in [dB]
$m_{index}$	modulation index
$N_0$	Noise Power Spectral Density; this notation breaks convention of using non-bold capitals for set scripts only
$\omega_E$	angular velocity of the Earth [rad/s] or [deg/s]
$\omega_F$	angular velocity of the satellite in the Earth Centric Fixed reference frame [rad/s] or [deg/s]
$\omega_s$	angular velocity of the satellite in the Earth Centric Inertial reference frame [rad/s] or [deg/s]
$p_{BER}$	probability of any bit in a bit stream being in error
$P_{PacketRx}$	Probability of Packet reception based on BER
$P_{rx}$	the received isotropic signal level at the ground station [dB]
$p_{rx}$	recieved power
$P_{tx}$	power of the transmitter [dB]
$\phi_i$	phase of signal of index i
$R^2$	coefficient of determination: an indicator of the of how well the data set is represented by the model

---



---

$r_{BER}$	Bit Error Rate for non-coherent AFSK modulation
$R_E$	radius of the Earth at 6378.14 [km]
$r_s$	radius from the centre of the Earth to the mean altitude of the satellite (assuming an eccentricity of approximately zero) [km]
$S$	the slant range from the satellite to the ground station [km]
$s_i$	output of transmitter index $i$ as a function of time
$SNR_{gnd}$	the ground station Signal to Noise Ratio power spectral density [dBHz]
$T_{txline}$	the ground station transmission line temperature [degrees K]
$T_{gndnoise}$	the ground station effective noise temperature [degrees K]
$T_{gndsky}$	the sky temperature according to the ground station [degrees K]
$T_{LNA}$	Low Noise Amplifier noise temperature [degrees K]
$\theta_{max}$	maximum elevation angle above the horizon from the ground station looking towards the satellite.
$\theta_{min}$	minimum elevation angle above the horizon from the ground station looking towards the satellite.



# 1 Introduction

## 1.1 Nanosatellites and CubeSats

Nanosatellites are small form factor spacecraft missions generally between one to ten kilograms in mass and operated in Low Earth Orbit ( $100 \text{ km} < \text{LEO} < 2000 \text{ km}$ ). These spacecraft typically use Amateur Satellite Service frequencies as defined by the International Telecommunications Union (ITU) for communications with ground stations [ITU2001, ITU2012, ITU2012a, ITU2012b].

Nanosatellites are often constructed by research institutions for technical demonstrations or low cost research platforms. Experiments on-board nanosatellite platforms have included the sensing of cosmic dust, Compton gamma ray observation, space tethers and biological studies. In 1999 the California Polytechnic State University at San Luis Obispo (also known as Cal Poly) along with Stanford University developed a standardized spacecraft structure and deployment system to allow universities to rapidly develop space systems without the need to additionally design rocket body integration systems or separation devices [Lee2015, Heidt2000].

Since the year 2000 over 2080 papers have been published related to the systems using the CubeSat spacecraft standard. Approximately 1260 of the papers are directly related to CubeSat communications systems. The number of articles illustrates the popularity of the Cube standard for the research community and the level of effort involved in the design of their communications systems.

One significant factor in the designs for nanosatellites is the availability of Commercial Off the Shelf (COTS) components that support the required radio frequencies. In this sense the selected hardware for communications systems evolves from the spacecrafts mission design process. Nanosatellite development can be characterized by short mission lifespans, rapid development, limited energy resources, low physical volume, fiscal constraints and, in the case of many universities, limited space experience. For example, the CubeSat specification limits the spacecraft to maximum of  $10\text{cm} \times 10\text{cm} \times 30\text{cm}$ . To facilitate quick development, students often cannibalize existing devices for components to build their spacecraft. This creative environment encourages nanosatellite developers to explore alternative options to traditional spacecraft subsystem design, in particular devices that combine micro-controllers, processors, modems and radio frequency transmitters within the CubeSat design constraints and mission requirements. As we explore the topic of CubeSat communications we will see how this methodology of using preexisting infrastructure can be applied to both the space and ground segments of the design.

## 1.2 Problem Statement

### 1.2.1 Motivation

Nanosatellites, and in particular the CubeSat standard, are emerging as a viable trend for space sciences both at the universities and in commercial space industry. Based on data tabulated by the Institute of Aeronautics and Astronautics from the Technische Universität Berlin (TUB) “TUB Small Satellite Database” (TUB SSD) [Buscher2014] approximately 287 different nanosatellites have developed by over 128 different student related institutions. Commercial interest in the platform is demonstrated by a company called Planet Labs, which announced in 2013 that it would create a constellation of 28 Earth-observation satellites using the CubeSat standard [Wall2013]. The company’s launch plans include a 131 CubeSat constellation by some time in 2015 [Taylor2014]. In 2010, Bouwmeester and Guo [Bouwmeester2010] provided an excellent summary of the CubeSat community’s missions at that time, for the journal *Acta Astronautica*. Their paper “Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology” illustrates the portion of successful CubeSat missions and their objectives. The authors surmised that over half of all pico- and nanosatellites launched have been in the CubeSat form factor with the single unit design being the most popular. As of July 2016, the website Celestrak.com operated by Dr. T.S. Kelso, reports 96 CubeSats actively being tracked by the Joint Functional Component Command for Space (JFCC Space) as a part of the United States Strategic Command (USSTRATCOM).

However, the development time required to create a viable communications link from the satellite to the ground (downlink) is prohibitive, especially for universities with limited budgets and human resources. Any time taken to design, build and test the communications infrastructure required cannot be used to develop the scientific payload for which the spacecraft is needed for in the first place. This represents a loss in development time for systems not directly related to the nanosatellite’s primary mission. Additionally the digital communications from the satellite must be supported by a terrestrial ground station requiring further development time and cost or be contracted out. By designing the on-spacecraft communications system to utilize existing terrestrial infrastructure the development time and costs for a nanosatellite could be reduced.

Commonly, university CubeSat link designers would use a spreadsheet, such as the one created by Jan King (W3GEY) in 2001 called the *AMSAT-IUAR Annotated Link Model System* [King2008]. This extensive spreadsheet includes many of the details needed to calculate an appropriate link budget and has been used by a number of university CubeSat projects [Castello2012, Schroer2009]. It is used to calculate if the satellite’s output signal could be interpreted on the ground. If the output value was low, in-depth knowledge of the system process was required so that combinations of variables could be altered to determine if the results were more favorable. Traditionally, for institutions new to the development of nanosatellites, the approach for designing the communications link from the spacecraft to the ground station on the surface has been largely manual and iterative. Additionally, this iterative process is premised on a large number of assumptions about the available transmission frequencies, the usable modulation types, the applicable baud rates, available energy for transmission and the available orbital slots. In particular this spreadsheet style of iterative

design and analysis does not account for variations induced by a CubeSat's orbital path. This last point is of key interest as it heavily influences the available time for communications.

Of interest to the university community is the development and implementation time required to deploy a communications system for a CubeSat.

As the variety and capability of COTS subsystem components increases a critical disparity arises. On February 25th, 2013 Surrey Satellite Technology Ltd (SSTL) deployed a CubeSat that used a Google Nexus One smartphone as the central avionics sensors for operations [Liddle2011]. The company had additionally planned to demonstrate the smartphone's use as the spacecraft's primary On-Board-Computer. NASA Ames has developed a program called PhoneSat, which successfully launched three CubeSats based on the same smartphone architecture [Alexander2013]. This particular smartphone came equipped with a 1GHz Qualcomm Scorpion CPU and Qualcomm Adreno 200 GPU. It supported up to 32GB of storage via microSDHC. Onboard sensors included a 3-axis accelerometer, ambient light sensor, digital compass, and a 5.0-megapixel camera.

Based on this COTS equipment let us explore a hypothetical systems design. If a single RAW12 uncompressed image is desired for maximum data extraction it would occupy 7.9 MB. If we assume that a CubeSat communications system designer is using one of the conventional data rates for Amateur Radio based communications at 9600 bits/s it would take 109.722 minutes to transfer this single image to scientists at the ground station. The preceding is only possible if the satellite were hovering over the ground rather than in a state of perpetual falling towards the horizon. If we assume the orbital altitude for the CubeSat is approximately 700 km this implies that the mean time the satellite has line of sight over the ground station is approximately 9 minutes for selected orbits. This would require 122+ orbits, accounting for passes which do not have line of sight to the ground station, to receive the uncompressed data. Of the orbits only roughly four per day will pass over the ground station if it is located along 49 degrees latitude. In combination, these factors result in a download duration of 30.5 days for a single RAW12 image. Obviously compression is required, but such methods are limited in their scalability.

Based on these components it is possible for a CubeSat to collect more raw scientific data than is possible to transmit to their designated ground stations within the lifespan of the spacecraft. The available time to transmit this data becomes one of the most precious commodities of the CubeSat design. It is obvious that on-board processing and data compression can play a role. However, compression can only play a limited role as it will imply that more science data will be stored in memory rather than transmitted to the ground. We can deduce that the future limitation of the CubeSat standard's contribution to science is the low line of sight access time regardless of the data rates of the communications systems used.

## 1.2.2 Problem Definition

Based on what has been described within the Sec.1.2.1 the total line of sight access time for a given CubeSat needs to be improved. An expanded terrestrial infrastructure beyond a single ground station is required. To accomplish this, a parametric study of the most influential factors for CubeSat link budgets is required.

### 1.2.3 Proposed Solution

Any solution to the problem of improved access time to the ground station network should combine the following requirements:

1. Commonly implementable equipment standards.
2. Commonly implementable software and protocol standards.
3. Readily available transmission frequencies.
4. Widely distributable, ideally taking advantage of existing wireless and wired networks if available.
5. Associated with activities outside of limited community of space communications to leverage the network effect of external communities.
6. Ideally the solution should accommodate future CubeSats as part of a linked constellation to enable other institutions to optimize the volume of their received data.

This thesis proposes to use the existing terrestrial Amateur Packet Relay System (APRS) I-Gate network as an example of how a virtualized ground station network could be used for CubeSat downlinks.

## 1.3 Thesis Formulation

### 1.3.1 Aim of the Thesis

The aim of this thesis is to determine if the preexisting terrestrial focused decentralized APRS I-Gate network can be utilized as a virtualized ground station network antenna array for CubeSat downlink.

### 1.3.2 Core Concepts within the Thesis

This thesis provides a parametric study of the most influential design factors for CubeSat link budgets.

Furthermore, this thesis shows that to effectively increase the available downlink time for a CubeSat within the current suite of design practices, it is necessary to shift to a distributed virtualized ground station network design. Such a system must integrate with the wider community's needs by providing them with a utility that is not necessarily correlated with the mission requirements of the CubeSat itself. Fortunately, this added utility for the community does not necessarily imply a radical change in radio design for the CubeSat itself.

### 1.3.3 Thesis Statements

This thesis demonstrates:

1. That while the classic technique of using Jan King's well developed AMSAT-UK spreadsheets [King2008] for CubeSat Link design allows for final tweaking of the parameters it does not allow for a parametric study that includes Orbital Mechanics or Operational planning with a needed degree of fidelity. As a result this classic technique does not allow for the more comprehensive measures of performance needed to compare link budgets as they relate to the initial mission design.
2. That it is possible to create generalized set of CubeSats models that include both the telecommunications systems and the orbital mechanics of those satellites. This may mitigate much of the initial assumptions and guess work for space systems designers.
3. That a generalized set of CubeSat models can provide actionable measure of performance relative to a virtualized ground station network.
4. That techniques of Short Time Window Analysis of Non-Stationary Time Series using classic methods does not statistically differentiate between a known stationary time series and a known non-stationary time series.
5. That the field of econometrics provides a very credible statistical method, the Phillips-Perrons test, for determining the stationarity of a time series signal. This is demonstrated using a known stationary time series and a known non-stationary time series.
6. That it is possible to increase the total volume of downlink data from a CubeSat by co-opting the capacity of distributed nodes, such as APRS I-Gates, into a virtualized ground station network.
7. That by using a virtualized ground station network for CubeSat downlinks the total received signal Error per Bit relative to Noise can be considered stationary when the sampled series length is between 250 to 650 data points.

### 1.3.4 Scope of the Thesis

The core body of this thesis will focus on the nature of CubeSat downlinks. The problems related to satellite uplink are easier to solve. For a given uplink it is possible to readily change the transmitter power, available equipment, and location. For obvious reasons this is not the case for downlinks due to their orbit. Thus, the domain of downlink design has a more pressing need for solutions.

### 1.3.5 Research Questions

1. Can the terrestrial APRS network receive a CubeSat transmission enable received packets?
2. If a CubeSat transmits on the common APRS network frequency can it cause rebounding packets, flooding back and forth across the network? Can this be prevented?

3. At each altitude category, determined by the CubeSat altitude histogram, what is the mean BER at the receiving I-Gates across North America if the satellite has 1W of transmitter output power?
4. At each altitude category determined by the CubeSat altitude histogram, provided a CubeSat with 1W of transmitter output power, is it possible for receiving I-Gates to get a signal greater than the predetermined Figure of Merit?
5. What is the maximum altitude (the realistic maximum spreading distance) that can be utilized by a CubeSat with 1W of transmitter output power and still have a signal greater than the Figure of Merit?
6. What could be the maximum volume of data downlinked by a CubeSat for a variety of modulation types and baud rates, within the amateur radio community standards?
7. What could be the maximum amount of data downlinked by the current CubeSat constellation? In other words, what is the volume of data the total population of currently active CubeSats could downlink via the proposed solution?
8. What is the optimal number of ground stations for the current constellation of CubeSats for the North American hemisphere?
9. What does the use of a virtualized ground station network provide that was previously not possible or available?
10. How can CubeSat transmissions be deconflicted to minimize the number of nodes interfering with terrestrial stations? Without FDMA (Frequency Division Multiple Access) since this is a single channel network? With limited TDMA (Time Division Multiple Access) because the orbits determine the initial time windows? With limited SDMA (Space Division Multiple Access) since the orbits determine the footprint or line of sight? With limited Code Division Multiple Access (CDMA) as the I-Gates would have limited *a priori* knowledge of the satellite transmitters and thus not know the required values for decoding?
11. How can we maximize the number of CubeSats that can use the network via the proposed solution?
12. Given the orbital elements of a satellites, the list of ground stations with their associated locations, could a CubeSat use this knowledge to address packets to the I-Gates which would incur minimal Doppler shift and longest viewing time, thus enabling the maximum total data downlink with minimal BER?
13. Does the use of this overlay network enable Quality of Service (QoS) measurements based on preformed packets?
14. With a QoS measurement can the downlink be provide with such that a cognitive radio capability to alter its transmit power and route packets through specific nodes to allow for energy savings?



15. How can we estimate the SINR encountered by the APRS I-Gates relative to our desired CubeSat signal?

### 1.3.6 Thesis Organization

This thesis is comprised of several chapters beyond this Introduction.

In Chapter 2 we will discuss the various aspects of not only amateur radio packet communications systems and satellite link design, but also orbital mechanics. This chapter describes many of the fundamental technologies and principles needed to understand the system design as we move forward.

Within Chapter 3 we will explore the design of the proposed solution. This chapter describes, at the overview level, the systems chosen along with the tools and methods used. Furthermore it describes how they interconnect at the logical level. In particular we will see how sets of models were created which are brought together into a simulation which provides a realistic measure of the network's performance and capabilities. The selection of models was influenced by surveys of the amateur community.

In Chapter 4 we see the specifics of how the simulation was set up to bring the system design to fruition. Within this chapter we will discuss how why the various models were created. Much of the supplementary information for this chapter is included in the Annexes. This includes example code, formulas, reference tables and figures.

In Chapter 5 we begin to distinguish between engineering and science for this thesis. Where the preceding chapters focused on answering the questions related to "is it technically possible to use a virtualized ground station network from APRS I-Gate nodes?", Chapter 5 focuses us towards measuring if the implemented system has real utility.

Finally in Chapter 6 we discuss the overall impact of the thesis and the system design.



## 2 Background and Literature Review

The goal of this chapter is to ensure that the reader is provided with the adequate background to understand not only the problem space encountered by the thesis but the underlying principles and technologies that are involved.

As such this section is designed to bring the reader up to speed on several key topic areas of the thesis.

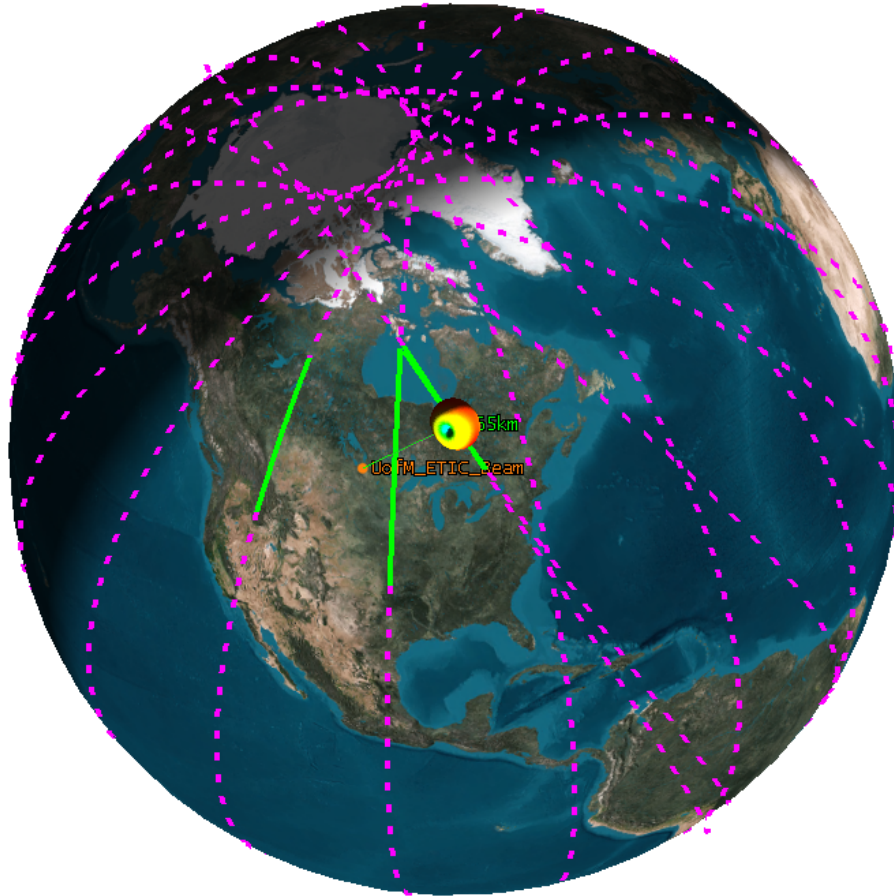
1. A brief introduction to the nature of satellite communications
2. An appreciation for the constraints imposed upon CubeSats with regards to communications
3. A brief introduction to distributed satellite downlink/uplink networks for design comparisons
4. A description of CubeSat design constraints as they relate to communications links.
5. An examination of the Amateur Radio Packet Reporting System which we intend to explore as an illustrative example for a virtualized ground station network
6. A literature review of the population of CubeSats looking specifically at their orbits and communications systems designs
7. An introduction and evolution of a generalized model for CubeSats called ExampleSat
8. Explore of the potential metrics of performance for our proposed satellite communications links

Overall one should think of the contents of this chapter as an exploration of the concepts and details required for the development of models to be used within constructive simulations for a proposed network of virtualized ground stations. Ideally the principles and models used within this chapter could be applied generally for development of other CubeSat models and related simulations.

### 2.1 Satellite Communications Networks

One of the key factor that impacts the capability of a scientific CubeSat's ability to downlink its collected data are the limited windows of opportunity to transmit this data to the ground station. In order for any data transfer to occur, the satellite needs to be in line of sight (LOS) with the ground station. LOS is dependent on two other factors. 1) the orbit that the satellite occupies and 2) the position of the ground station or stations.

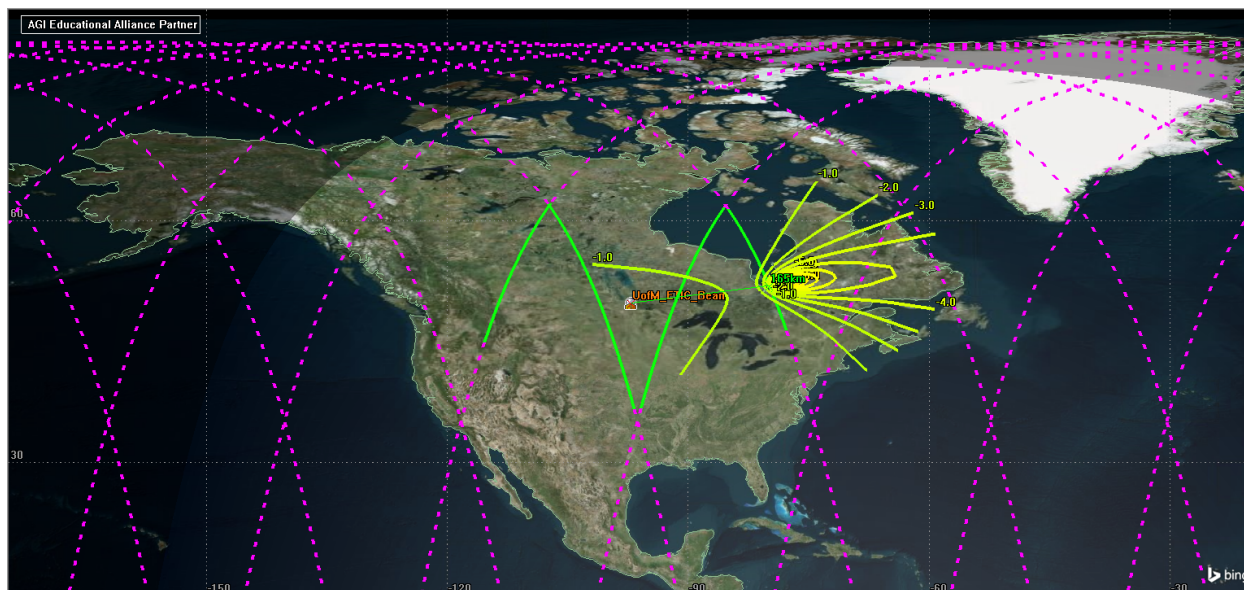
In Fig. 2.1 we see a typical CubeSat downlink illustrating the scale of the problem space. The colourful toroid shape represents the satellite's far field antenna emission pattern, also called the beam pattern, assuming a half-wave dipole antenna is used. The dashed magenta lines represent the CubeSat's orbital path. Green dashed lines indicate durations when the satellite has LOS to the ground station. In this simplistic example, for the CubeSat to successfully communicate with the ground station the main lobe of its antenna far field radiation pattern should be aligned coincident with ground station.



**Fig. 2.1:** STK scenario of CubeSat downlink.

This leads to the topic of access times. Access times, also known as access windows, are the periods in which the ground station has LOS with the spacecraft. Based on the satellite's current position (true anomaly) within its orbit, the LOS is limited by the time at which the elevation angle from the ground station is above the horizon. All of this relative to the ground station's position as illustrated in Fig. 2.2. In this picture the dashed magenta lines are the orbital path and the green lines are periods of LOS with the ground station. Yellow contour lines representing the received signal strength, in dB, from the satellite to the Earth's surface.

In this example scenario the satellite is in a sun-synchronous orbit (it crosses the poles and has a precession of one degree per day) with an altitude of 165 km. This altitude represents the lower median currently occupied by a portion of the CubeSat population. By running a



**Fig. 2.2:** Model scenario of a satellite at 165Km altitude, sun-synchronous orbit and the resulting LOS from the U of M.

simple simulation Analytical Graphics Inc.'s Systems Tool Kit (STK), such a satellite yields the following access times over a 24 hour period.

**Table 2.1:** Access times for the example satellite shown in Fig. 2.2.

Access	Start Time (UTCG)	Stop Time (UTCG)	Duration (seconds)
1	21 Oct 2015 18:28:46.421	21 Oct 2015 18:34:57.316	370.894
2	21 Oct 2015 19:56:35.755	21 Oct 2015 20:01:10.796	275.041
3	22 Oct 2015 05:52:23.738	22 Oct 2015 05:58:52.395	388.657
4	22 Oct 2015 17:56:20.752	22 Oct 2015 18:00:00.000	219.248
		Min Duration	219.248
		Max Duration	388.657
		Mean Duration	313.460
		Total Duration	1253.841

As seen in Table 2.1 over a 24 hour period the ground station could at best communicate for a total of 20.9 minutes. At worst the could contact the satellite for only 3.65 minutes. This scenario also does not include the typical minimum elevation angle of ten degrees which would further restrict the access times. Depending on the CubeSat's altitude, access time can be a significant limiting factor for the spacecrafts overall utility.

### 2.1.1 Example LEO Satellite Communications Downlink Networks

To appreciate the difficulty associated with retrieving data from a satellite via its associated ground stations, we will now look at how selected example networks approach the problem of access times.

### 2.1.1.1 NASA NEN

A traditional approach to solving the limited transmission window time is to increase the number of ground stations. A prime example of this approach is the NASA Near Earth Network (NASA NEN) as shown below in Fig. 2.3.



**Fig. 2.3:** NASA Near Earth Network is an example of multiple redundant ground stations [SCaN2014].

This network is comprised of a number of ground stations geographically distributed across the globe, increasing the likelihood that any LEO satellite that requires Tracking, Telemetry and Command (RT&C) has LOS with a viable connection to the network. The network utilizes a number of large diameter dishes capable of tracking a spacecraft across the sky. This system design enables robust if not assured communications between the terrestrial controllers and the spacecraft.

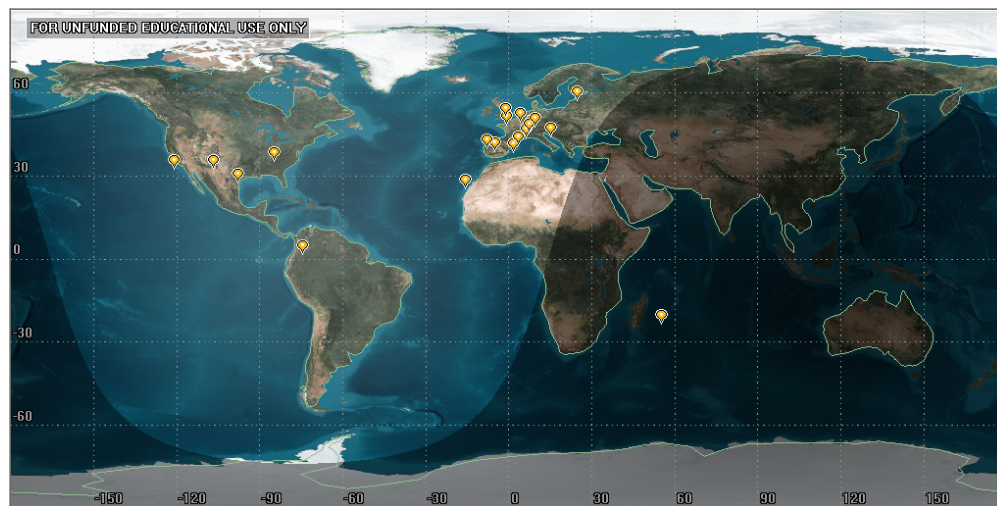
Unfortunately for smaller institutions such as universities, access to such a network is limited. Viable alternatives to this professional common user ground station network are in development. This leads us to example networks such as Global Educational Network for Satellite Operators (GENSO).

### 2.1.1.2 GENSO

The GENSO project [Shirville2007, Leveque2007] uses a similar solution to the above mentioned NASA NEN. This large scale endeavor seeks to combine the equipment of universities,



research institutions, space industry partners and private citizens into a large scale redundant network of centrally coordinated ground stations. However, these institutions use large diameter dishes such as those shown in Fig. 2.3. Universities typically use some combination of Yagi-Uda directional antennas with tracking rotors.



**Fig. 2.4:** Unconfirmed Global Educational Network for Satellite Operators (GENSO) Ground Station Servers (GSS).

To function properly, this laudable effort requires coordination between many different bodies in different countries to apply new equipment and new software standards. As of July 29, 2016 it is not clear how many working antenna systems are a part of the network. Each ground station functions as both uplink and downlink for connected spacecraft, with a central server coordinating global schedules. From Fig. 2.4 we see a clustering of ground stations based near European institutions.

Ostensibly this common user ground station network design claims that by participating in the network a satellite can increase its communications time from 3% to 60% at low cost.

### 2.1.1.3 FUNcube

The FUNcube project [Shirville2016] is a joint team of volunteers in the United Kingdom's AMSAT-UK and the Netherlands AMSAT-NL creating a series of CubeSats. The original goal was to design, build and launch a single satellite. They have since created four follow on space systems related projects. AMSAT is the collective term for amateur radio satellite organizations around the world which design, build, procure launch and operate satellites that have amateur radio related payloads [AMSAT2016]. OSCAR is series of satellites created by AMSAT.

FUNcube uses a unique method of downlink from the spacecraft. The team developed a USB connectable Software Defined Radio (SDR) that members of the public could purchase. The latest version (available at [www.funcubedongle.com](http://www.funcubedongle.com)), as of July 2016, is called the FUNcube Dongle Pro+ and is used for communicating on the LF and L bands from 150 kHz to 1.9 GHz with a gap between 240MHz and 420MHz [Long2011].



**Fig. 2.5:** FUNcube. Photo from Wikimedia Commons (Photo taken by Wouter Weggelaar).

Since the ground link hardware is publicly available across a radio spectrum wider than their CubeSat's original design, the dongles are appealing to potential users outside the typical space community. The developers of the satellite can sell to enthusiast users around the world whom will download the applicable software package, connect the USB dongle, connect an appropriate antenna and then act as a downlink site for their satellite on 145.935 MHz at 1200 baud BPSK (Binary Phase Shift Keying) FEC (Forward Error Correction) modulation. The software loaded by users onto their computer includes the capability to adjust for Doppler shift induced by the spacecraft's movement. As an inexpensive and "hackable" system one appeal of the FUNcube dongle is its ability to be used for purposes other than to receive signals from FUNcube. Users have created SDR radio telescopes [Morg2013], a system for visualizing amateur radio signals [Whatton] and Weak Signal Propagation Reporting for digital modes using very low power [Garratt2013].

As such the FUNcube's model of ground station distribution has been very successful and the FUNcube dongle is supported by a variety of hobbyist SDR projects.

By providing the community with a common interface and making it an extensible platform the FUNcube project has been able to expand their ground station network far beyond the traditional communications link design. In comparison with GENSO, FUNcube's downlink network is more widely distributed across a number of users, while the uplink is limited to selected stations with dedicated hardware. We can speculate that FUNcube's ground station does not use dishes as shown for NASA NEN, but that it may be similar in principle to the Yagi-Uda array designed for SwissCube as shown in Fig. 2.7. The tracking antenna illustrated in Fig. 2.7 is pointed at the satellite as it passes overhead. This is typically performed by





**Fig. 2.6:** A FUNcube Pro USB Dongle. From: Wikimedia Commons (Photo by Jeff Keyzer).

automation using software that predicts the spacecraft's orbit.

The principle concept of the FUNcube network is used as an abstraction for the network design of this thesis. Although it was not possible to ascertain the number of connected ground stations within the FUNcube network, its success in leveraging an existing community's capabilities to create an overlay network is worth consideration and adoption.



**Fig. 2.7:** Example AMSAT/OSCAR Yagi-Uda antenna ground station antenna. From: Wikimedia Commons SwissCube Ground Station (Photo by Olivier Chatelain).

## 2.2 ExampleSat-165

Throughout the remainder of this thesis we will use a theoretical example CubeSat rather than the generic term. This will aid in discussion later as we examine specific alterations to the spacecraft and its associated mission design. This method of referring to a specific theoretical spacecraft rather than the generic was used to great effect in Larson *et al.*'s Space Mission Analysis and Design [WileyJ.Larson2004].

As a convention we will name our theoretical satellite ExampleSat. Hyphenation and numbers following the proper noun "ExampleSat" will refer to the mean altitude of the orbit above the Earth's surface. We will assume that ExampleSat-165 has nearly circular orbit that has a mean altitude of 165 km above the Earth's surface. The mission design parameters of ExampleSat-165 will be expanded upon as we further discuss its requirements. In particular we will provide more definition and details for ExampleSat-165 in Sec. 2.5 as we explore the population of CubeSats and their orbital parameters. For now, our notional ExampleSat-165 only requires a name and justification for its suffix.

## 2.3 Virtualization and Overlay Networks

### 2.3.1 Network Virtualization

The paradigm of virtualization suggests that "*if you do not have what you want you can virtually create what you want with what you have*"[Sitaraman2014]. Originally this concept of virtualization allowed for the creation of virtual rather than actual systems or devices for use within the computer science and information technology domains. In 1975 Gagliardi provided insight into the growing trend of virtualization through his paper "Trends in computing-system architecture"[Gagliardi1975] in which he predicts the evolution of virtualized computational subsystems, storage subsystems and communication subsystems. The key takeaway in Gagliardi's paper is that the data storage methods and the individual user processes that utilize this data are separated. This concept is enabled by the rapidly decreasing cost of processing technology and storage media. Data storage and user processes can employ different optimization techniques due to their different hardware/software requirements. Virtualization allows both the data storage and the user process to evolve and expand using different technologies or methodologies, while still enabling them to work together. We should note however that Gagliardi's paper is limited in its discussion of virtualization of communication and applies the concept at what is now the equivalent of the Open Systems Interconnection model (OSI model) Data Link (frame) and Network (packet) layers, which hadn't been described at the time of his paper.

The current generation of virtual machine software packages such as those provided by Oracle VM Virtual Box in which operating systems can be nested inside one another, allowing an Microsoft Windows user to run a Linux operating system without requiring a second physical computer. This scenario allows for the separation of the software from the underlying hardware resources. Hardware virtualization involves the abstraction and sharing of hardware resources among different parties. Virtualization has become a key component in cloud-computing. As an active area of research, the application of virtualization to wireless

communications can enable flexible sharing and reuse of the existing infrastructure, thus lowering capital expenditures [Wen2013].

Since virtualization can involve the sharing of hardware resources among different parties, classic overlay networks could be considered a limited form of virtualization.

### 2.3.2 Overlay Network

In simplistic terms an overlay network is an additional layer of connectivity built on top of one or more existing networks. In this classic sense the original Asymmetric Digital Subscriber Line (ADSL) is an overlay network that enables high speed data communications over twisted pair copper lines without disturbing the original voice telephone network.

In 2014, Sitaraman *et al.* suggested that overlay networks could be used to alleviate outages, congestion, lack of scalability, and slow adaptability in their paper titled “Overlay Networks: An Akamai Perspective [Sitaraman2014].” Although their paper is focused on resiliency of intent services, the capabilities that Sitaraman *et al.* describe for overlay networks are similar in nature to problems faced by satellite ground station downlinks. For satellite networks, outages are created when the spacecraft is out of LOS with any of the ground stations. Scalability is also an issue in that the distribution of useful ground stations is limited by participating organizations and their locations as well as the cost of building specialized ground station equipment. Using a dedicated ground station network as described in Sec. 2.1 can also lead to slow adaptability where newer CubeSats use older telecommunications designs to connect with aging terrestrial infrastructure.

Based on the limited capabilities of most individual educational institutions to create purpose built ground stations, we can safely assume that a network virtualization schema of some variety as outlined in Sec. 2.1.1 is desired. Notionally we will use the existing Amateur Packet Reporting Systems (APRS) network as a example network on which to study the utility of an overlay network for CubeSat downlinks. We will investigate using the APRS network to create a virtualized ground station receiver that is distributed, at scale, across North America.

## 2.4 Automatic Packet Relay System

The protocols and communications design (Connectionless Emergency Traffic System: CETS) that would eventually become APRS was initially created in the 1982 by Bob Bruninga (WB4APR) to map high frequency US Navy position reports as part of his role as a research engineer with the United States Naval Academy. From its early stages, APRS was designed as a robust protocol for digital communications exchange between a number of geographically dispersed nodes. Unlike most packet radio systems, APRS operates in an unconnected broadcast fashion in which packets are not acknowledged nor are re-transmissions requested for lost packets. All of the nodes within a geographic region, in this case North America, all transmit and receive packets using the AX.25 protocol on the same frequency of 144.39 MHz [Loveall2005]. The result is a robust and streamlined protocol that has unique features and disadvantages for a communications link designer.

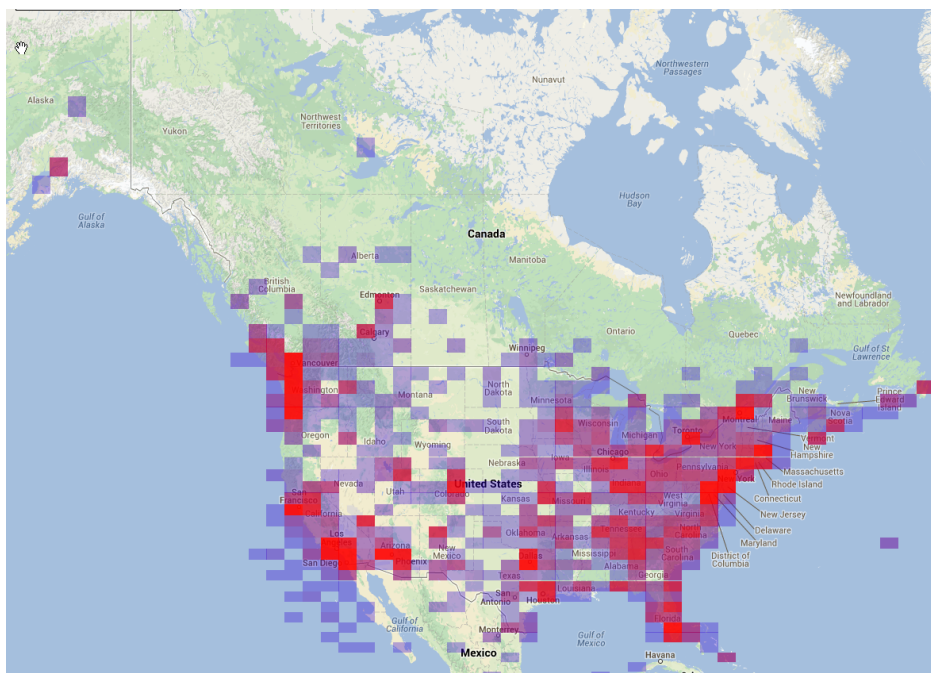
In his 2014 masters thesis with the California Polytechnic State University campus in San Luis Obispo, Kenneth W. Finnegan (W6KWF) strove to resolve a number of inconsistencies with, and the overall lack of, documentation on APRS [Finnegan2014]. His thesis starts with the history of APRS and ends with an analysis of the network's RF channel capacity. Since APRS was created prior to the development of the OSI model, Finnegan's efforts to describe APRS within the context of modern communications design is appreciated.

If the node is primarily use to beacon out its call sign and position, possibly with another small line of text, such nodes are colloquially called a tracker. Nodes can receive packets from other nodes and repeat them, with amendments to the routing information traveled by the packet, expanding the reach of the network. Such nodes are referred to as “digipeters”.



Finally, nodes can act as a gateway between the RF network and the internet at large. This last type is called an I-Gate (Internet Gateway) and is of particular interest for our concept of virtualizing ground station receivers. For this research, the objective is to simulate the transmission of RF packets from ExampleSat-165 to the I-Gates so that the data is collated by servers available on the internet. Additionally, the design should avoid being relayed through digipeters since a large number of nodes could receive ExampleSat-165's packets and thus "bounce" between Digipeters flooding the network with excess messages.

APRS could be considered a ubiquitous network across the North American amateur radio community. Fig. 2.9 shows the commonality of APRS nodes distributed across North America over a period of one hour.



**Fig. 2.9:** APRS node “heat map” of North America. Red indicates a higher density of nodes. From: APRS.fi

Because of its geographic ubiquity, ease of connectivity and open standards, the APRS network represents a good candidate for the creation of an overlay network to virtualize the needed ground stations for CubeSat downlink communications. The large number of ground stations may increase the total downlink connectivity time during an orbital pass of the satellite. The wide geographic dispersion will limit the amount of expected outage time when the CubeSat is not within LOS of a ground station. The APRS network also represents an advantage in scalability since the network effect of adding new nodes to the existing set of connected stations can also benefit APRS enabled CubeSats. Furthermore, as the radio frequency spectrum for APRS is dedicated for amateur radio communications, interface sources are limited.

For CubeSat link designers a key point is that the infrastructure is already in place. It has been assembled over the years by a community of enthusiasts for a purpose other than CubeSat communications. Any licensed amateur radio operator is freely allowed to

contribute to the network. CubeSat designers that would want to communicate using the amateur radio bands still have to apply for licensing through their respective governmental bodies. They must also apply for Satellite communications spectrum via the International Telecommunications Union just as any other space systems designer. However, there is spectrum set aside specifically for amateur radio satellite systems.

Before using the APRS network as a notional virtualized ground station network, we must also consider that there are three significant obstacles to be overcome. The first is to determine if packets sent by CubeSats would be capable of being received by APRS network nodes. The APRS network was designed for digital communications between nodes on the Earth's surface moving at limited speed if at all. This network was not designed for communication with nodes moving at approximately eight (8) km/s at a distance of 2000 kilometers. The second obstacle is to determine if such received packets would have insurmountable errors, either from individual ground stations or from the cumulative set of stations receiving packets at the same time. The third issue is that any use of the protected radio frequency band, and use of the APRS network itself should not interfere with the amateur radio community's original use of the allocated spectrum nor the APRS network itself. If we transmit on 144.39 MHz any APRS node across North America could hear us.

A key consideration of the APRS network is its heterogeneous decentralized structure. Much like the internet, many users bring their own hardware and software together aligning it with a common set of standards in order to create a robust and useful network. Like the internet, this infrastructure already exists and is in day to day operation. As we have seen uses for the internet evolve beyond its initial design, the proposed virtualized ground station network concept extends the use of APRS beyond its initial design context.

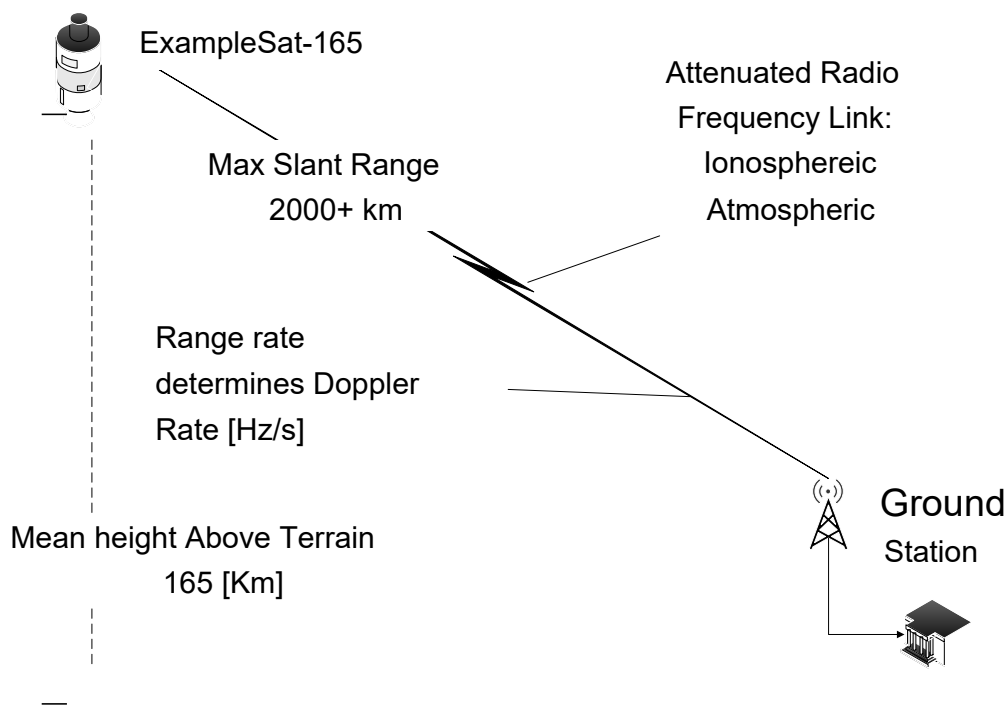
### 2.4.1 Space vs Terrestrial Communications

To appreciate the issues associated with using the APRS network for our virtualized ground station network we need to investigate the nature of satellite ground station downlinks.

The diagram in Fig. 2.10 shows the physical layout of the problem space associated with ExampleSat-165's communications. The slant range shown in the figure changes as the spacecraft travels at approximately eight (8) km/s depending on the altitude in Low Earth Orbit. In this simplistic example we can consider the CubeSat to be traveling along a vector that would pass directly overhead of the ground station. This orthogonal alignment of spacecraft and ground station or a target of interest is sometimes called nadir. The rate of change in range to the ground station induces Doppler shift compared to most terrestrial radio applications.

From Fig. 2.10 we can learn a number of details that make radio communications with ExampleSat-165 unique compared to terrestrial radio links. First is that the mean altitude of ExampleSat-165 above the terrain is not directly the primary concern, but does give rise to the slant range from the satellite to the receiving ground station. This slant range is dependent on a number of factors such as the orbital position of ExampleSat-165 at that instance of time relative to the ground station's geographic location on the Earth's surface. Note that the antenna system in Fig. 2.10 is most likely a mechanically tracking antenna system similar to the one depicted in Fig. 2.7.

Next we need to compare how Fig. 2.10 differs from the typical APRS network topology.



**Fig. 2.10:** Physical diagram of ExampleSat communications downlink.

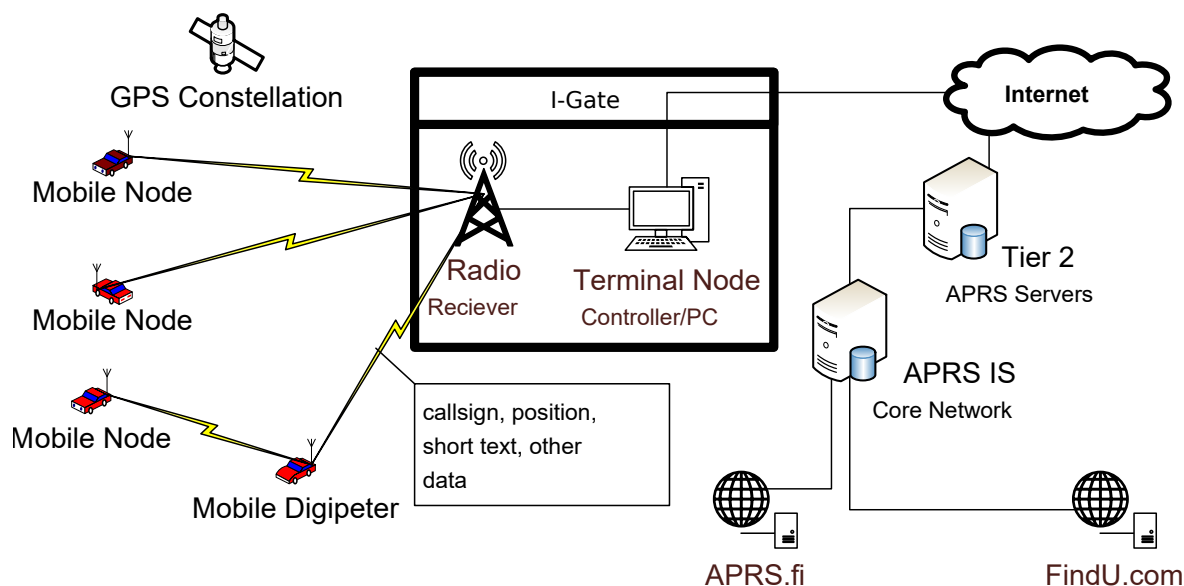
The diagram in Fig. 2.11 shows a typical network topology for APRS nodes. Each node connects via Radio Frequency from the mobile nodes to a fixed station called an I-Gate. Next the RF packet is converted through either a Terminal Node Controller device or other hardware or software solution (such as a sound card) through an internet gateway to eventually reach the APRS Internet-Service (APRS-IS) Core Network. From here, the data is provided to selected internet websites for services such as finding a particular call sign's location on a map as provided by APRS.fi.

Now that we have an appreciation for the scale of the APRS network and are beginning to see the scope of the problem from a satellite communications perspective we can see how modelling and simulation is required to investigate the systems design in greater detail. If we plan to model and then simulate how the APRS network could receive APRS packets from orbit, we need to explore the nature of the modulation used by the network as they will effect the success of the downlink.

### 2.4.2 Audio Frequency Shift Keying (AFSK) Modulation

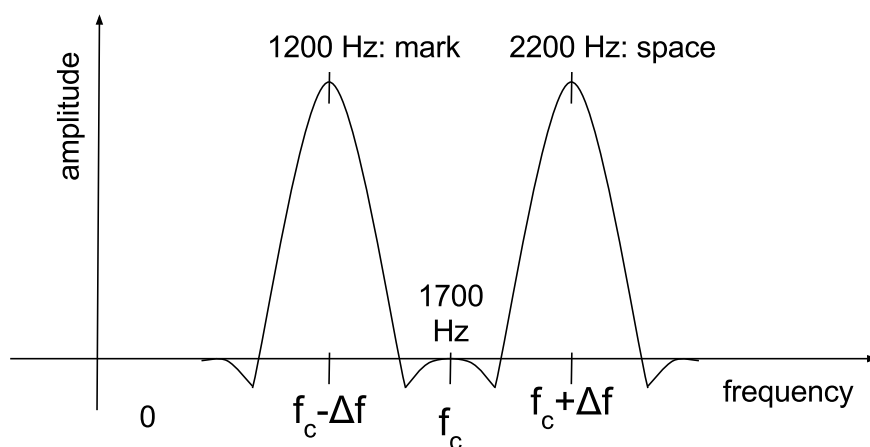
To encode digital information on the analog FM signal used for APRS the Audio Frequency Shift Keying (AFSK) modulation technique as described by Scott Miller [Miller2016] on his N1VG.net website. While creating an open source tracker project, Scott Miller was disappointed at the level of documentation available for low level details on packet radio. This modulation technique was adapted for amateur radio purposes as it was popularly used within modems based on the Bell 202 standard. Bell 202 uses a 1200 Hz tone for marks





**Fig. 2.11:** Physical & logical diagram of the APRS Network.

(ones) and 2200 Hz for spaces (zeros) as shown in Fig. 2.12.



**Fig. 2.12:** AFSK audio baseband modulation frequency distribution.

Because the tone frequency is within the audible range this allowed amateur radio enthusiasts to attach a Bell 202 based modem to the microphone and speaker of their radios to begin experimenting with digital operations. Effectively the digital data passed through the audio components of the radio normally intended for voice communications. The modulated signal could “pass through” AC-coupled devices allowing for a very wide variety of radio equipment to be utilized for digital packet communications. Notably, the AFSK also uses Non-Return to Zero Inverted (NRZI) encoding such that if a zero (0) is transmitted as a change in tone, a one (1) indicates no change in tone.

The Bell 202 standard uses one of two tones to modulate data onto baseband signal, so we may be able to consider it, and thus AFSK, to be a form of Binary Frequency Shift Keying (BFSK) using base band audio signals rather than the typical radio frequency signal. However, this design choice leads to limited efficiency, requiring higher transmission power and more overall bandwidth.

At this point we must look at the terminology commonly used within the amateur radio community as it relates to modulation and data. Their common terminology differs from what is used within technical literature. Settings for radio equipment such as those depicted in Fig. 2.14 often, and continue to use the term baud (Bd). As defined by Federal Standard 1037C of the U.S. national Communications System [NCS1997] the baud (Bd) is a unit of signaling speed equal to the number of discrete signal conditions, variation or events per second. Baud itself is actually a unit of modulation rate, which is the rate at which a carrier is varied to represent the information in the digital signal. Remember that the information rate and the modulation rate are not always equal. In a bit stream, the bit rate is the number of bits occurring per unit of time, typically in bits per second.

Furthermore, if the signal transmitted within each unit interval can be of any  $n$  discrete states, the bit rate is equal to the rate in baud times  $\log_2 n$ . For the case of our AFSK modulation there are two distinct states, thus the multiplier is one (1) so the bit rate and the baud are equivalent. The term baud is largely deprecated within modern telecommunications systems, but settings within amateur radio equipment that are APRS capable still use it.

It instructive to examine the bauds used and thus the bandwidth and modulation index used by AFSK. These details will lead to more capable and accurate models and/or simulations depending on the software tools used.

The common bauds used for APRS are 1200 and 9600 with the former being the most prevalent for terrestrial applications within the on 144.39 MHz within North America. We can see in Eq. 2.1 that the relationship between the signaling rate to bit rate for AFSK.

$$r_{signal} = \left( \frac{f_b}{N} \right) \quad (2.1)$$

Where  $f_b$  is the bit rate and  $N$  is the number of bits encoded into one signaling element. Since  $N=1$  for binary Bell 202 tones the baud is equal to our bit rate.

Based on the bit rate we determined the Bandwidth ( $w$ ) used in Equation Eq. 2.2 can be formulated:

$$w = 2 (\Delta f - f_b) \quad (2.2)$$

Where  $\Delta f$  is the frequency deviation from the centre carrier frequency such that:

$$\Delta f = |f_m - f_s| \quad (2.3)$$

Where  $f_m$  is the mark frequency (binary 1 at 1200 Hz) and  $f_s$  is the space frequency (binary 0 at 2200 Hz). Within the baseband signal the centre audio frequency is 1700 Hz  $\pm 500$  Hz.

To fit AFSK within a simulation software tool, its modulation index  $m_{index}$  may be required:

$$m_{index} = \frac{|f_m - f_s|}{f_b} \quad (2.4)$$

Applying these formulas according to the baseband signal as AFSK does there are two options. 1) At 1200 Baud the modulation index  $h$  is:

$$m_{index} = \frac{|f_m - f_s|}{f_b} = \frac{|1200 - 2200|}{1200} = 0.833$$

2) At 9600 baud the modulation index works out to:

$$m_{index} = \frac{|f_m - f_s|}{f_b} = \frac{|1200 - 2200|}{9600} = 0.104$$

By comparison, and only for illustrative purposes, the same process within the RF spectrum rather than the baseband signal are:

$$m_{index} = \frac{|144.3888 [MHz] - 144.3922 [MHz]|}{9600} = 2.833 \times 10^{-6} \simeq 0$$

and at 1200 Baud:

$$m_{index} = \frac{|144.3888 [MHz] - 144.3922 [MHz]|}{1200} = 3.54166667 \times 10^{-7} \simeq 0$$

The above values are unrealistic at these bauds since a systems designer would likely use a higher baud available at such higher frequencies, but it does show the change in efficiency do to the selection of the audio domain as the baseband signal.

The binary combination of 1200 Hz and 2200 Hz in AFSK baseband modulation can be heard as the audio tone that screeches or howls when listening to the received FM audio signal. From a radio frequency perspective, there is no difference between an analog FM voice broadcast and our digital AFSK signal. Effectively the “voice” in the case of AFSK is a rapidly switching audio tone.

Based on interviews and a literature review conducted by Finnegan [Finnegan2014] in December of 2014, we can suggest that the FM deviation for AFSK for APRS is 3.5 kHz for both the 1200 Hz and 2200 Hz tones. Starting with a frequency of 144.39 MHz it becomes 144.39 MHz -3.5 kHz for a 1200 Hz binary one and 144.39 MHz +3.5 kHz for a 2200 Hz binary zero. An in depth study of an optimized choice FM carrier deviation is not required within the scope of this thesis as our overall goal is not to optimize the existing network but to bootstrap our overlay network onto it. The 3.5 kHz deviation is commonly used by the amateur radio community and for our comparative purposes could even be arbitrary. Finnegan also mentions that the classic Bell 202 signal for a basic 3002 telephone channel (for which Bell 202 was designed) should maintain both the 1200 Hz and 2200 Hz tones with a  $\pm 10$ dB window.

### 2.4.2.1 G3RUH FSK

A competing standard called G3RUH FSK, developed by James Miller (call sign G3RUH)[Miller1988], further advanced the use of APRS for satellite communications. Using a 9600 baud it uses audio responses from 10-20Hz up to 5kHz. An alternative baud of 19200 is also available. G2RUH method requires that no distortions in amplitude or phase are present within the total audio frequency spectrum used. The result is an output spectrum from 145.009850 MHz to 145.010150 MHz for a nominal operating frequency of 145.01 MHz.

James Miller's work is used by the amateur radio community for a variety of projects including satellite communications and was under consideration for QB50 project [Weggelaar2006, Bedon2010], which seeks to combine 50 university built CubeSats into a lower thermospheric constellation [QB502016]. However, the premise of this thesis is to use the most commonly found terrestrial stations. In Sec.2.5 we will explore, based on data from a community survey, the most commonly used communications schema for APRS.

### 2.4.3 AX.25 UI Packet Format

To determine the capabilities of digital telecommunications system its important to not only understand the physical layers but also its data link layers. The APRS network operates on the AX.25 data link layer protocol based on the original ITU Telecommunications Standardization Section (ITU-T) X.25 [Beech1998]. Table 2.2 illustrates the layout of bytes within a Unnumbered Information (UI) frame for the protocol. With the table the abbreviations are listed as: Dest. Addr. is the Destination Address, Src. Addr is the Source Address. Ctrl Field is the Control Field, FCS is the Frame Check Sequence. The flag fields are used to indicate the beginning and end of a packet.

**Table 2.2:** AX.25 UI Packet Format in bytes for terrestrial APRS. After: [Lustig2015]

Flag	Dest. Addr.	Src. Addr.	Digipeter Addr.	Ctrl Field	ID	Info. Field	FCS	Flag
1	7	7	56	1	1	256	2	1

Note that the routing path (Destination Address) is optional for the case of AX.25 as used by APRS. In the particular case of higher altitude objects such as aircraft, weather balloons and satellites we don't want messages to flood repeatedly across large portions of the terrestrial network. A wiki page at Info.aprs.net had a well written description of the various path settings for most applications around the world. However, the original page is more easily accessed by the searching the Internet Archive for the previously mentioned URL. Of particular interest was the following quote:

#### “High-Altitude Balloon Stations

Do **not** set any path for balloons. With altitude, the beacons will be heard by stations over a very wide distance, eliminating the need for digipeating [*sic*]. Using a WIDEn-N path in this scenario will simply spam the frequency.”

In the normal scenario an alias is used in the Destination Address field such as WIDE1 - 1, WIDE2-2. At each hop within the network the suffix is deprecated or the address

field replaced with the repeating node's address until no more hops are available. By not including any addresses within the Destination Address field the packets can be received but not retransmitted by any digipeter over RF. Fortunately for the case of higher altitude objects such as balloons, aircraft or spacecraft, nodes such as I-Gates will still post all received packets to the core network.

To verify this functionality of the AX.25 aliasing feature, a simple test was performed using three radios. A Kenwood D700 was installed in a truck and set to have no path information for its outgoing packets. A Yaesu FTM400 Delta was installed in a car using the typical sparse node density settings of WIDE1 - 1, WIDE2-2. An existing APRS I-Gate with a GE MVS radio was also used. Beacons were manually and automatically initiated from the radio in the truck. None of these packets were rebroadcast by the digipeter in the car, but all of the packets were viewable on APRS.fi through the I-Gate.

Anecdotally while discussing this research with members of the community they expressed concern. It is often assumed that since the spacecraft would be at high altitude it would have a line of sight with most of the continent and thus flood the network with "bouncing messages" repeating between nodes. In effect, echoing across the landscape via RF repeaters. The very simple experiment performed above, illustrates how careful selection of path settings within the radio system can mitigate one of the most common complaints an amateur radio community could have about using the APRS network for high altitude or space experimentation. However, by removing the path information only the I-gates will repeat the messages, sending the contents to the Core APRS network servers.

Based on the simple experiment in digipeter addressing, we can redo Table 2.2 from a space communications perspective. The values shown in Sec. 2.4.3 represent the fields for terrestrial AX.25 encoded packet for APRS. By contrast, Table 2.3 shows how the fields should be filled when APRS is used for high altitude and space systems wishing to communicate via the I-Gate network.

**Table 2.3:** AX.25 UI Packet Format in bytes for High Altitude / Space. After: [Lustig2015]

Flag	Dest. Addr.	Src. Addr.	Digipeter Addr.	Ctrl Field	ID	Sci. Data	FCS	Flag
1	null value	7	null value	1	1	256	2	1

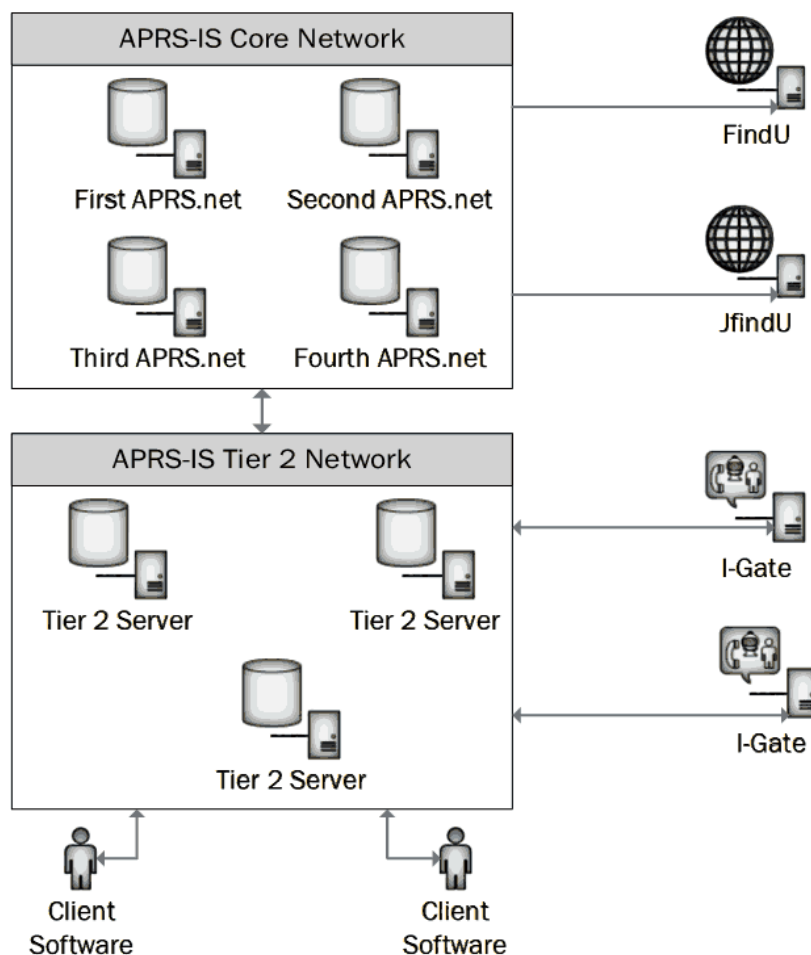
Combining what we know from Table 2.3 and a model of the orbital mechanics for ExampleSat-165 we can characterize the data channel between the satellite and the ground station(s). The orbital mechanics component will be further explored in Sec. 2.5.

We should also note that its possible to specify the destination address explicitly allowing for directed traffic to a node. Since there is no alias to deprecate nodes would not relay/repeat the packet and only the intended recipient node could act accordingly, and thus prevent flooding of relayed messages across the network.

### 2.4.4 APRS-IS Network Topology

Any data transmitted across the APRS network is only useful if it can be relayed to users in a timely and user friendly manner. Web services, such as APRS.fi shown in Fig. 2.8, rely on data fed to a common server network from APRS users across the globe. This service is commonly called APRS Internet Service (APRS-IS).

Due to the number of unique world wide APRS users sending data across the network on any given day, a coordination effort was required to enable higher order applications. According to aprs2.net server systems operators coordinated their efforts starting in the summer of 2000 with a tiered server architecture. The logical interaction between these tiers can be seen in Fig. 2.13.



**Fig. 2.13:** Diagram of the APRS-IS network. After: Geoff Gatward, VK2XJG [Gatward2009].

Note that within this network concept the I-Gates convert RF packets, transmitted by beaconing nodes, into TCP packets and post these to APRS-IS Tier 2 servers. This in turn is transferred to APRS-IS Core Network. Packets and messages can transit out from the APRS-IS Tier 2 network to the I-Gates, which would broadcast those packets out via the RF medium. Data sent to the APRS-IS Core network can be queried by web servers such

as APRS.fi for posting on websites and other services. Client software in this case could include the very popular UI-View [Barker2016] for Windows or Xastir [Mills2016] for Linux computers.

The Tier 2 network (T2) provides load balancing and any server outages are redistributed among the set. Additionally there are strict guidelines for operating a T2 server which includes the use of redundant power feeds. The Core network supports 16300 user connections, the T2 provides 55349 user connections [Pacier2016]. This level of capacity and redundancy would be impractical for a single university scale institution to create for a single spacecraft program, which provides another layer of reasoning for using APRS as virtualized ground station network.

### 2.4.5 I-Gate Hardware Design

I-Gates convert RF packets, transmitted by beaconing stations, to TCP packets and post these to APRS servers. This in turn is queried by websites such as APRS.fi for publicly viewable locations and other services. The I-Gates are built and operated by amateur radio volunteers and as such are heterogeneous in design and implementation. For example, its possible for an I-Gate to not only receive packets but also send them. This is typically not done to prevent flooding the local RF network area with packets from the internet. However, a fixed station such as an I-Gate may act as a beaconing station for regional information such as local news, emergency information or other information services.

A plethora of different radios and computer equipment are used by community members to create and operate an I-Gate as Fig. 2.14 shows. We can see in 2.14a a small form factor Raspberry Pi being used as the both the computer and internet connection while a Software Defined Radio dongle acts as the RF receiver. Fig. 2.14b illustrates a more classic design with a Kenwood TM-267 tuned to 144.39 MHz connected to a Kantronics PacketCommunicator 3 acting as the Terminal Node Controller (TNC). The TNC acts similar to a Packet Assembler/Dis-assembler with an additional modem that converts the baseband digital signals to audio tones to be modulated onto the RF carrier. Fig. 2.14c shows the Radio and TNC from Fig. 2.14b connected to a PC running UI-View to display a map of the APRS stations and any messages traversing the network.

We see in Fig. 2.14a a possible example for a physical wiring diagram of an I-Gate antenna design called ground plane antenna. The antenna systems utilized by the amateur radio community within the APRS network for I-Gates are most often omnidirectional in nature and emit a toroidal radiation pattern as shown in Fig. 2.15b. This maximizes the horizontal field to increase reception to other terrestrial nodes. The resulting 3D far field radiation pattern from that antenna at the given frequency of 144.39 MHz was modelled using software called 4nec2. The image in Fig. 2.15a shows a model of the antenna's wire elements. We will look at 4nec2 and its capabilities in greater detail as we model more of the I-Gate antennas.



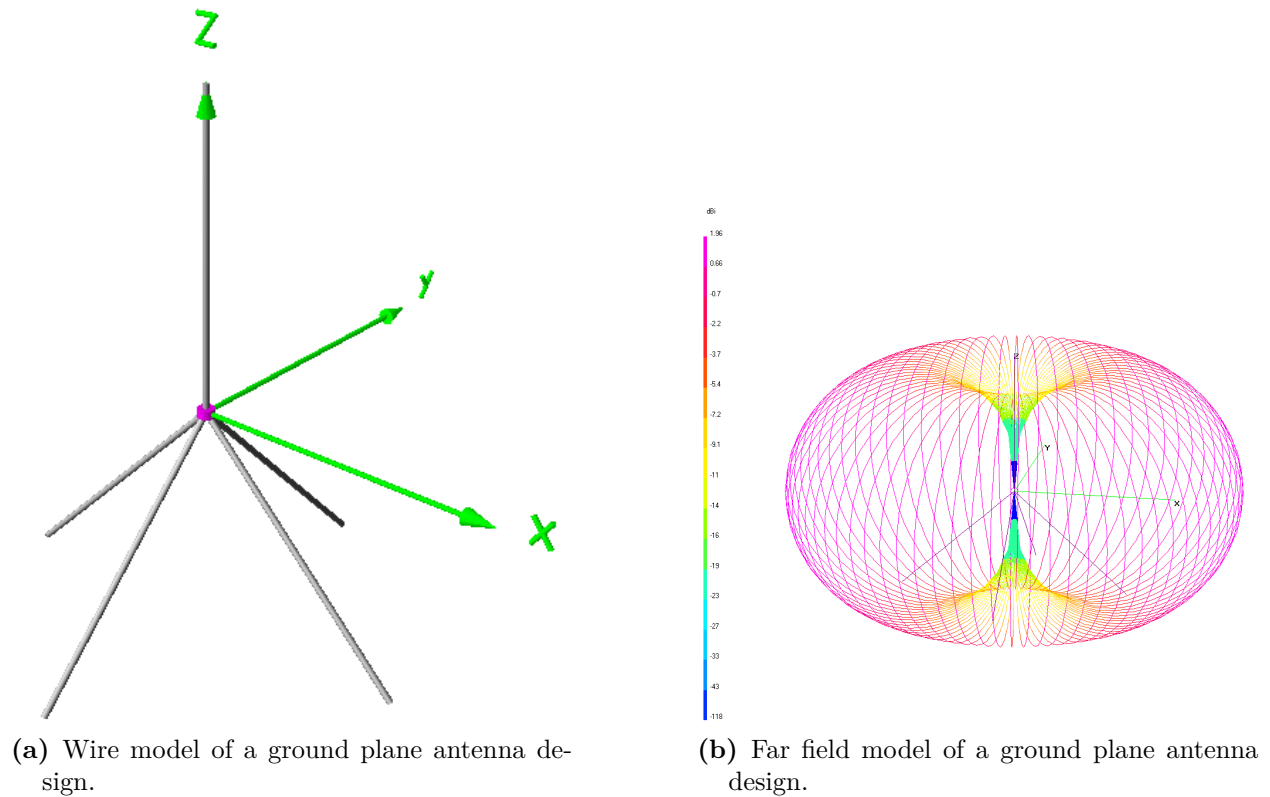
(a) Raspberry Pi I-Gate



(b) Radio and TNC I-Gate. From: VE4COR

(c) I-Gate as Sub-Figure b with a PC running UI-View.  
From: VE4COR**Fig. 2.14:** Different I-Gate station configurations.





**Fig. 2.15:** Example APRS I-Gate ground station antenna models in 4nec2.

We are interested in the far field radiation patterns for the antennas used by the ground stations because the apparent null or minimum gain area is at the top of the model along the vertical  $z$ -axis, orthogonal to the Earth's surface. Conversely the maximum gain for the antenna is along the circumference of the toroid. If we consider two terrestrial nodes attempting to communicate with one another using the APRS protocols, these antenna far field patterns are advantageous as the maximum gain for both antennas is applied across the horizon thus maximizing the gain of the communications link. However, the same antennas when viewed from LEO can be a potential problem for our virtualization of the I-Gate ground stations into an overlay network for ExampleSat-165. As the satellite passes overhead of a given ground station the orientation of this null will cause received signal from ExampleSat-165 to be severely attenuated. These factors must be accounted for in any models or simulations related to ExampleSat-165's communications link budgets.

### 2.4.6 APRS-IS Software/Firmware Design

Due to the heterogeneous nature of the hardware in use for the APRS network it would be impractical to discuss the software used in APRS I-Gates. This is especially true since some may use a PC with an audio card to conduct modulation and demodulation while others may use dedicated hardware. However, there are guidelines for any I-Gate to follow as best practices so as to not earn the ire of the community. In fact the APRS-IS.net landing page describes the behavior of APRS-IS nodes stating that "Because the primary function

of APRS-IS is to support **Amateur Radio APRS on RF** (sic), the following must be kept in mind by users and software authors who inject packets directly into APRS-IS:” [Loveall2016a] By injecting packets its presumed that Peter Lovall implies that the node is receiving RF packets to be demodulated and then sent of TCP/IP to the Tier 2 network shown in Fig. 2.11. He further suggests that packets are assumed to be gated from the RF domain. Linking further into the APRS-IS Server Design page the site defines what features a node must have to constitute an APRS-IS server, which allows all clients to receive all packets seen by the server. It is worth noting that duplication removal of packet occurs at the server level and is based on the origin call and Service Set Identifier (SSID), the destination call SSID, the data length and the data content. The SSID is typically based on the amateur operator’s unique call sign with an added suffix based on the application the APRS node is being used for. Furthermore duplication is done over a 30 second sliding window for each packet. After these checks are done the packets are eventually passed via TCP/IP to the Tier 2 servers and onward.

### 2.4.7 APRS “q-construct”

The transport layer functionality of the APRS-IS is implemented by a feature called the “q-construct” [Loveall2016] which adds series of conditionally specific characters to the data packet. For example “qAR” denotes that the packet was received directly from an I-Gate using the I construct, and that any call sign SSID after the characters qAR is the call SSID of the I-Gate itself. Other “q-constructs” include qAC for packet from a client node directly via a verified connection, qAX for the same but via unverified connections. This allows unverified connections generated by unlicensed stations to propagate through the I-Gate but are not rebroadcast by other types of nodes.

Steve Dimse (K4HG), the developer of FindU.com [Dimse2014], in personal communications clarified the nature of “q-construct”s and how this feature could be used to analyze the network. The “q-constructs” enable a TNC or other device/software to check the character strings within a packet and make transport layer decisions. These include APRS-IS network entry identification, loop detection, automatic loop protection, comparability with current I-Gates and client software.

The following was collected from APRS.fi in a query for the call sign VE3JJS during the experiment described in Sec. 2.4, where the transmitting node did not specify a destination nor alias in the address field for its packets:

Last position: 2014-11-20 04:20:51 UTC (11m ago)  
2014-11-19 22:20:51 CST local time at Kenora, Canada  
Course: 0°  
Device: Kenwood: TM-D700 (rig)  
Last path: VE3JJS>4Y2U2Q via qAR,VE3JJA

We can see that the path data shows the packet was transmitted by VE3JJS and was relayed to 4Y2U2Q via a direct confirmed path through the I-Gate with the call sign VE3JJA. As we will see in the Chapter 3 this qAR construct allowed us to isolate the call signs of I-Gates

from the rest of the APRS packets across the network and thus narrow down the set of I-Gate geophysical locations to create a network map of nodes.

## 2.5 CubeSat Population Survey

To create a representative model for the CubeSat's downlink, we need to study the communications systems used by these satellites and the orbits they occupy. In April of 2013 Klofas *et al.* [Klofas2013] wrote a paper entitled "A Survey of CubeSat Communications Systems" for the California Polytechnic State University. This excellent survey paper collates the various disparate sources for the description of CubeSat communications systems into one cohesive set. By combining all of these sources together the author's were able to make conclusions about the state of CubeSat communications systems as they existed at the time. Of particular note was their conclusion that CubeSats should include a simple beacon for tracking purposes. They also suggest using common communications modes to leverage the interest of the amateur radio community. This last critical detail relates to the need for external assistance to accomplish the spacecraft's mission objectives. Additionally they stressed the importance of monitoring and verification of the ground station link.

The core premise of this thesis relies heavily upon the work of Klofas *et al.* [Klofas2013] as a basis for the initial model design. By parsing the data provided by their ongoing work (most recently updated to the 8th version of their survey on August 8th 2014) we could determine some critical trends in the communications design of the current generation of nanosatellites. We should note that the Institute of Aeronautics and Astronautics from the Technische Universität Berlin (TUB) publishes a spreadsheet called the "TUB Small Satellite Database" (TUB SSD) [Buscher2014]. The database uses a number of different online sources to collect its details about both nano- and pico- satellites, including Klofas *et al.* The TUB Small Satellite Database describes the country of origin, the operator, the purpose, the dimensions, the launch details, the bus power, and information about the downlink. However, we will use Klofas *et al.* as our representative sample of spacecraft for several reasons. The TUB database includes spacecraft in the much broader category beyond CubeSats, some of which have diverse sizes and shapes. This would imply different solar energy collection capabilities and thus widely different transmitter power capabilities. This in turn would also lead to a larger set of antennas designs onboard the various spacecraft. The Klofas *et al.* data set can be considered a contiguous set of satellites loosely fitting a more similar design constraint in that each fits within the 1U to 3U CubeSat specification which allows for comparison between spacecraft using the following parameters:

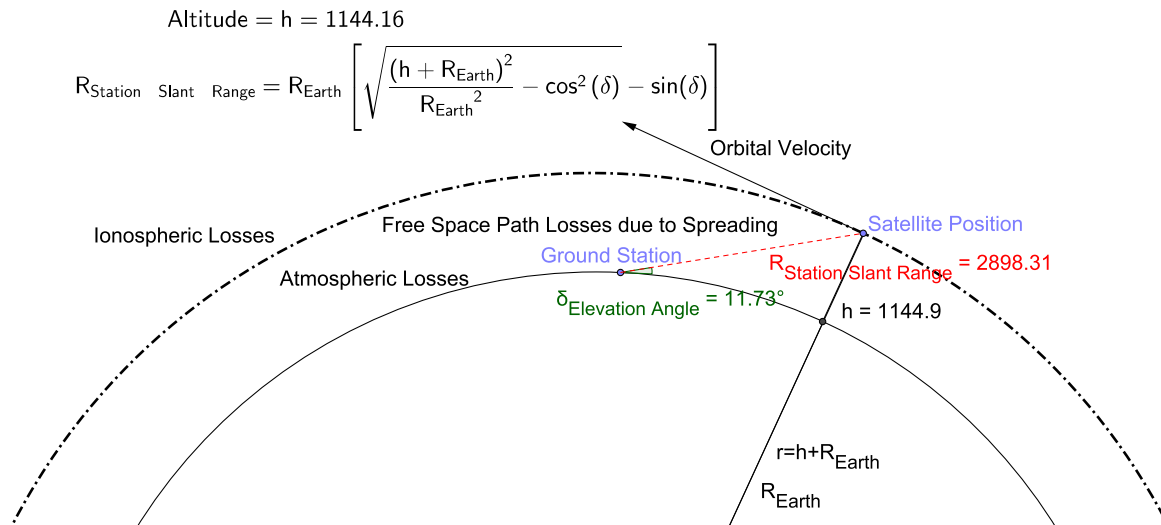
- Transmit power
- Downlink frequency
- Protocol used
- Data Rate
- Modulation type

- Antenna Type
- Object Name and Number

By looking at the population histograms describing the popular systems designs within communications systems in Klofas *et al.*'s survey we can create a representative model that can be used as a generalized but realistic representation of a CubeSat. This data is used to complete the parameters for ExampleSat-165 and its iterations if required.

### 2.5.1 Orbital Design Impacting Telecommunications Design

From Fig. 2.16, we can surmise that spherical spreading loss is the largest source of signal degradation that can impact ExampleSat's ability to successfully communicate with a ground station. In this section we will examine the orbits chosen by CubeSat designers to see if any trends are apparent.



**Fig. 2.16:** Downlink with associated losses. After:[King2008].

One key utility given by Klofas *et al.* and TUB SSD's surveys were the name and object number (NORAD Catalog ID) for the various spacecraft. By tallying these object identifier codes we input these IDs into a web service called Space-Track.org which provides Two Line Element (TLE) sets that describe the orbital trajectory of each object. This service is operated by the JFCC Space within the United States Strategic Command as part of NORAD. This data was also corroborated with Celestrak.com's current NORAD elements for CubeSats. Celestrak.com is a website operated by Dr. T.S. Kelso whom is instrumental in the space community. His website provides a number of services which include a list of CubeSat missions and their orbital elements [Kelso2015]. This was required as some of the names listed by various projects do not match their official identification. Additionally not all of the CubeSats found in Celestrak's data set were listed with Klofas *et al.*'s tables. To further advance the research within the CubeSat community a robust and volunteer centralized database of CubeSat specifications would be extremely helpful.

To create a set of generalized CubeSat models we must account for the orbits these spacecraft occupy. If the majority of CubeSats are within a narrow set of orbits, we can adjust our model accordingly without the need for arbitrary assumptions.

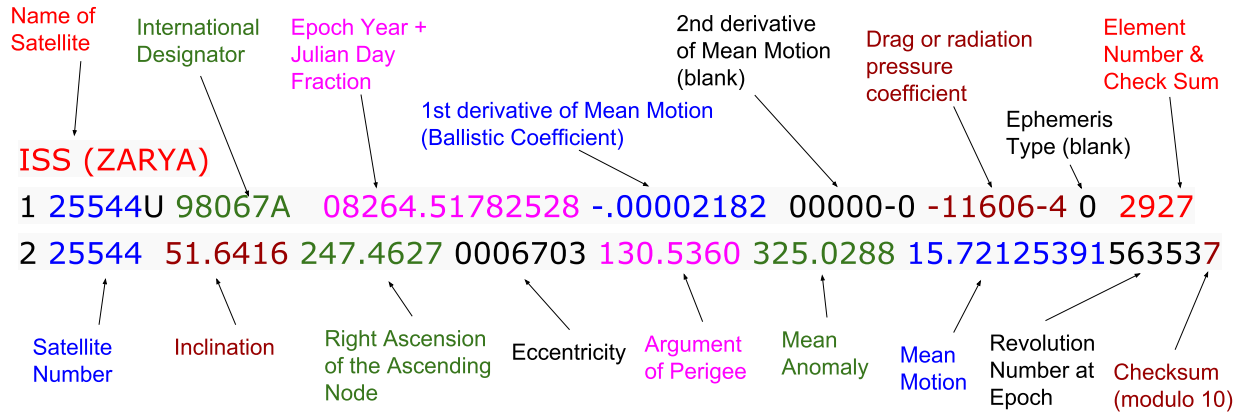
The TUB SSD [Buscher2014] includes some orbital parameters (apogee altitude, perigee altitude, eccentricity, inclination and period) within its data set. Recreating the orbits, and thus doing comparative studies of CubeSat communications systems including coverage and revisit times, the epoch time also needs to be included. As Earth orbiting satellites are often described from an inertial frame of reference the epoch time defines the starting point for Keplerian calculations of that orbit. Evolution of those orbital elements due to gravitational bodies, non-sphericity of the earth, atmospheric drag, radiation pressure and other forces can then be applied as perturbations to the model for that specific orbit of that spacecraft. For follow-on research we should note that each of the “real time” orbital elements drifts with time due to these perturbations and is updated by the JFCC Space Surveillance Network (SSN) either once per week or per month depending on the orbital altitude, the spacecraft size, and any anticipated orbital conjunctions. Since the TUB SSD is distributed as a spreadsheet it will lag the current positional information, thus if it were to include the epoch time it would enable valid analytical comparisons.

### 2.5.1.1 A brief introduction to orbital mechanics

To appreciate how issues such as slant range described in Fig. 2.16 are influenced by ExampleSat’s motion as it orbits the Earth we must understand the basics of orbital mechanics. For descriptive purposes we will use the TLE mentioned previously. The TLE is a data format used to encode a list of orbital elements for any Earth orbiting object at a specified point in time called the epoch. If provided a predictive formula or model, one can use the TLE to determine the position and velocity of a space object at any point in the past or future within a margin of error. The TLE is used specifically for a set of orbit propagation models called Simplified Perturbations Models (SPM) of which there are five different mathematical models. These models predict the perturbations to a space object’s trajectory based on the Earth’s shape, the object’s aerodynamic drag, and gravitational effects such as attraction to the sun or the moon. For example, in 2009 Nicholas Zwiép Miura’s masters thesis was a comparative study of the Simplified General Perturbation Model (SGP4) and software from NASA’s Orbital Debris Program Office [Miura2009]. The original reports regarding SGP4 are available from Dr. TS Kelso’s Celestrak website [Hoots1988]. The SGP4 model is as popular as it is used with the common TLE format of data shown below.

For building a generalized model of CubeSat orbits we can extract the inclination, eccentricity, and mean motion from the set of CubeSat TLE’s and compare them. If a significant portion of the population uses similar orbital elements we can create categories of CubeSat orbits. Since the eccentricity determines how elliptical an orbit is it therefore determines how varied the mean altitude above the Earth’s surface and thus the variance of slant range.

The TLE differs from the classic Keplerian elements as they are designed to be used with an orbit propagator. The commonly used terminology for discussion of orbital mechanics elements are the eccentricity, semi-major axis, inclination, RAAN, Argument of Perigee, true anomaly (spacecraft position) and finally the epoch time. For completeness and future reference all of the TLE fields are described here:

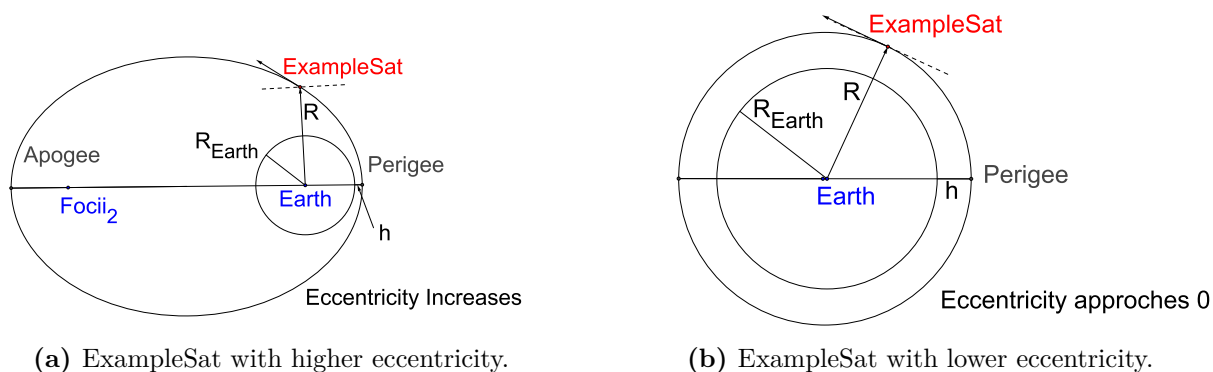


**Fig. 2.17:** Example TLE. After: NASA [NASA2011]

- Name of the space object.
- Satellite Number used in the USSPACECOM catalog to designate the object. The “U” is for unclassified.
- International Designator which includes the launch year, the 067 in this case the the 67th launch of that year. The “A” tells us its the first object from that launch vehicle.
- Epoch Year as the reference year.
- Julian Day Fraction as the fractional number of days passed in the particular year, in this case the epoch year 2008. Combined the Epoch Year and the Julian Day Fraction create a reference point in time for all the time varying elements in this TLE set.
- First derivative of the mean motion which is the daily rate of change of revolutions completed each day, divided by two.
- Second derivative of the mean motion.
- Drag term representing the radiation pressure coefficient.
- Element Set Number & Check Sum used by USSPACECOM as a running tally of all TLE generated by its catalog service.
- Inclination of the angle between the equator and the plane defined by the orbit.
- Right Ascension of the Ascending Node (RAAN) which is the angel between the ver-nal equinox an the Northbound point where the orbit crosses the equatorial plane (ascending node).
- Eccentricity which determines the circular shape of the orbit from 0 to 1.
- Argument of Perigee which is the angle between the ascending node and the orbits point of closest approach.

- Mean Anomaly which is the angle between the perigee and the satellite's current location in its orbit referenced to a circular orbit with an equivalent semi-major axis.
- Mean Motion which is the mean number of orbits the object completes in a day.
- Revolution Number and Check Sum which is the orbit number at the Epoch Year for this TLE set.

For this thesis the relevant TLE fields include the Name, Satellite Number, Epoch Year, Julian Day Fraction, First derivative of the mean motion, Inclination, RAAN, Eccentricity, Argument of Perigee and Mean motion. However, the software system STK, used to create the final discrete event simulation within this thesis, can utilize a complete TLE set directly to simulate a spacecraft's orbit.



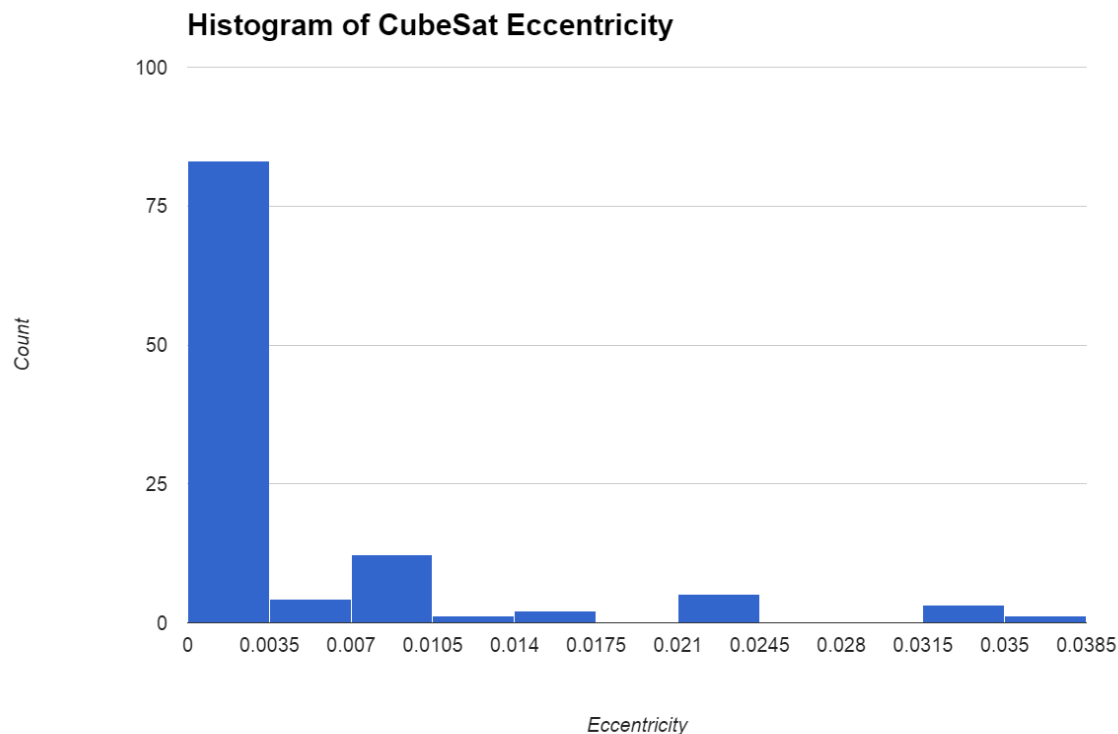
**Fig. 2.18:** Eccentricity as it impacts ExampleSat's slant range.

As we can see in Fig. 2.18 the eccentricity of the ellipse describing the orbit's path around the Earth determines the range of altitudes for ExampleSat. If the distribution of eccentricities for the population of CubeSats has a wide variance from zero (0) to one (1) then it would not be possible to create any categorizations of ExampleSat that would maintain any semblance of generality.

## 2.5.2 CubeSat Orbits

Based on the CubeSat mission names from Klofas and TUBS [Klofas2013, Buscher2014] and data collected from SpaceTrack.org we can see from Fig. 2.19 that the population of CubeSats are within a circular orbit as their eccentricities approach zero.

If we then look at the distribution of inclinations for the population of CubeSats we see that a large portion of are within 96 to 104 degrees. Since the inclination is the angle between the orbital plane and the equator this implies that many of the CubeSat are in polar orbits. When we also examine the histogram of altitudes as shown in Fig. 2.21 a pattern emerges. Given the limited range of eccentricities, proportion of inclinations and the altitudes used we can state that CubeSats occupy circular Low Earth Orbits (LEO) that traverse the geographic (true) poles.



**Fig. 2.19:** Histogram of CubeSat Orbit Eccentricity. Data derived from Klofas *et al.* [Klofas2013].

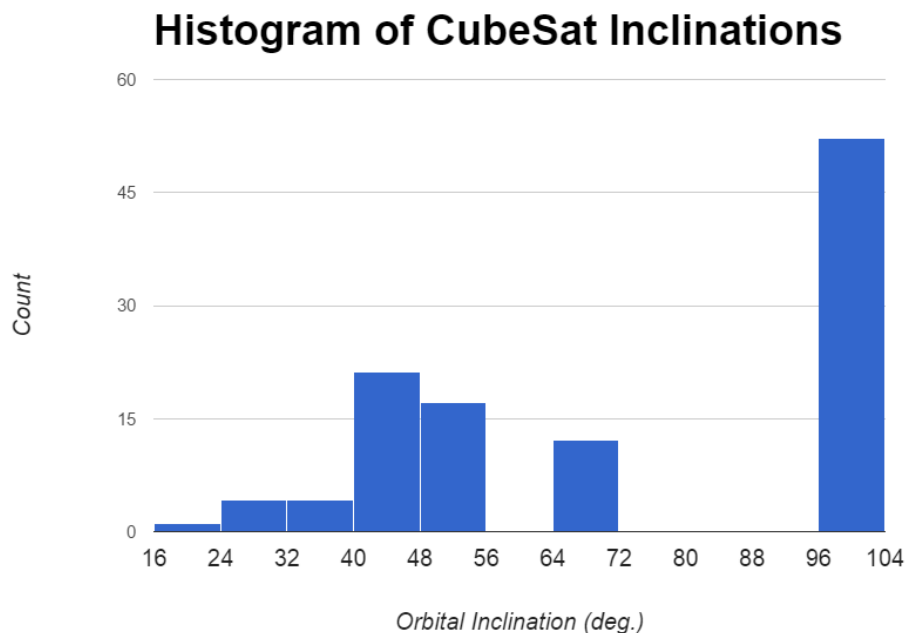
From a mission design perspective this makes sense intuitively. CubeSat missions have traditionally been platforms for scientific measurement and/or technology demonstration for several reasons. The volume of a CubeSat at increments of 10 cm x 10 cm limits the spacecraft’s ability to collect solar energy through deployed solar arrays. This in turn limits the available energy for RF transmission. With less available RF power the satellite could not overcome the large spreading loss associated with longer slant ranges due to higher altitude orbits.

As the altitude of an orbit rises its mean motion slows and each orbit takes longer to complete. Thus at higher altitudes a CubeSat would also spend more time in eclipse as it passes from the sun side across the terminator between day and night as shown in Fig. 2.22. This would in turn increase the need for more energy storage batteries which increases mass, cost and complexity.

The CubeSat standard was designed such that the nanosatellites are used as ballast mass on launch vehicles for primary payloads heading into orbit and thus typically rideshare on the launch vehicle. As such the orbit requirements of the primary mission payload of the launch vehicle determines the orbit available to the CubeSat mission.

CubeSats with Earth observation missions have similar sensor requirements for global coverage at “close” range. Conversely, CubeSat missions with technology demonstrations or experiments as their purpose may or may not require specific orbits. All of this explains why the population of CubeSats occupy a narrow selection of orbital elements; they often



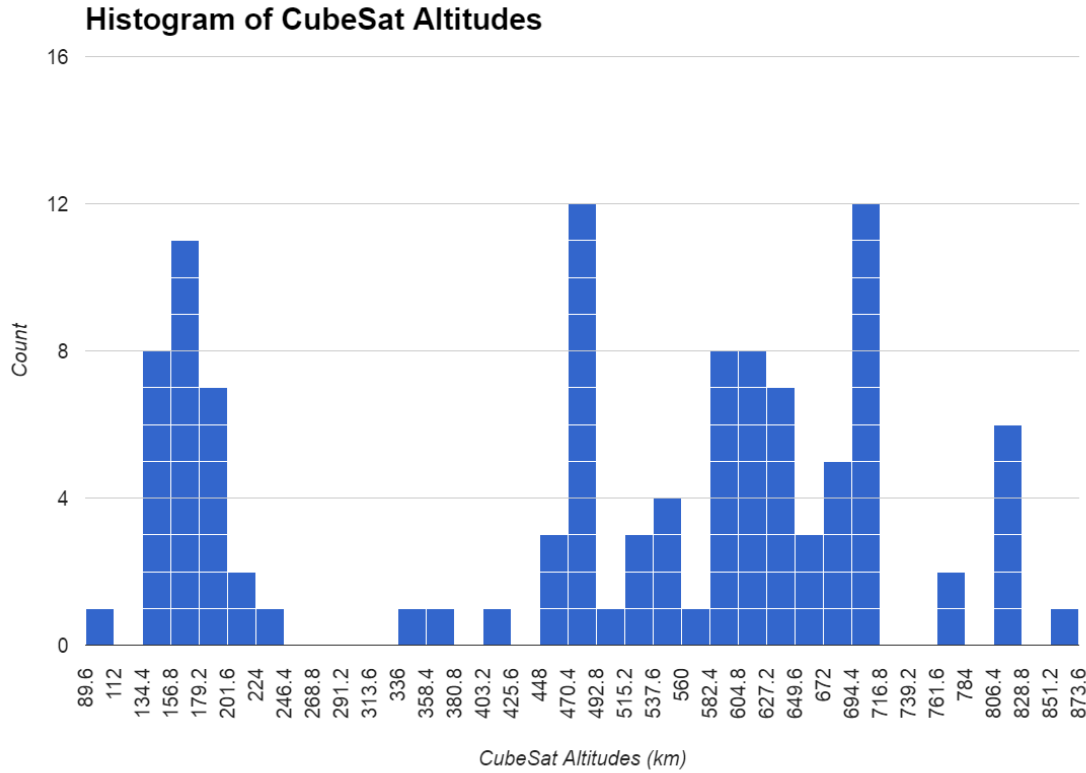


**Fig. 2.20:** Histogram of CubeSat Orbit Inclinations. Data derived from Klofas *et al.* [Klofas2013].

“hitchhike to space” with other Earth observation primary payload and benefit from the same orbit. The Annual Small Payload Rideshare Symposium was created to coordinate the small payload community[SPRSA2016]. In 2012 Micheal Swartwout from Saint Louis University expanded on his previous work with “A Statistical Survey of Rideshares (and Attack of the CubeSats, Part Deux) [sic]”, in which he states that although rideshares are only 7.5% of all payloads, “the bulk of new manifests will go to CubeSat-class spacecraft [Swartwout2012].”

All of what we explored so far within the orbital elements for the population does not describe a satellite’s orbit in its entirety as does a TLE. However, since we are interested in communications systems design for ExampleSat-165 we can make certain assumptions. Since the eccentricity approaches zero the apogee (point of closest approach) and perigee (point of furthest approach) distances should be similar if not equivalent. Thus the argument of perigee has little bearing on our generalized ExampleSat and can be arbitrary. Both the mean anomaly (true anomaly) that describes ExampleSat’s current position will be arbitrary as will epoch time. These arbitrarily chosen orbital elements have little impact on the communications capabilities of a spacecraft with a circular orbit. Note that if the eccentricity or inclination changes both can impact the communications system.

Notice that in Fig. 2.20 there is a cluster of satellites which includes 51.6 degrees. This could be a byproduct of the International Space Station’s NanoRacks Smallsat Deployment program where CubeSats are ferried up to the ISS as cargo and jettisoned off the Japanese Kibo module arm [NanoRacks2016]. Note that for such lower inclinations the satellite may not achieve a line of sight that includes more extreme latitudes, thus limiting the spacecrafts



**Fig. 2.21:** Histogram of CubeSat Orbit Altitude. Data derived from Klofas *et al.* [Klofas2013].

operational coverage area.

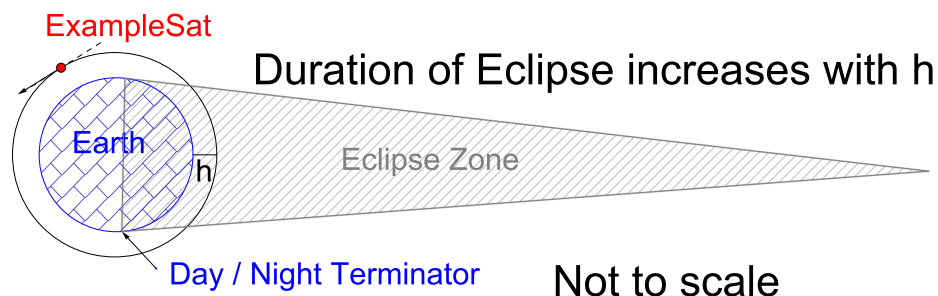
Based on the data shown in Fig. 2.21 we can create a list of potential altitudes, shown in Table 2.4 for ExampleSat's orbital altitude and if refer definition of ExampleSat in Sec. 2.2, the suffix to identify it specifically. This also allows us to determine the orbital period of the satellite.

**Table 2.4:** Potential ExampleSat altitudes (suffixes) based on Fig. 2.21.

Median Altitude (km)	Orbital Period (min)
165	91.83
480	98.34
605	100.97
715	103.30

From what we have explored we can create a basic list of orbital elements for ExampleSat-165:

- Apogee (and perigee) altitude of 165 Km
- Inclination of 97.7924 degrees based on the median between 96 and 104 degrees



**Fig. 2.22:** ExampleSat eclipse zone with Earth's day / night terminator.

- Eccentricity of 0 degrees
- Argument of Perigee of 0 degrees
- RAAN of 204 degrees
- Starting True Anomaly of 0 degree

We can see how this orbit will appear by inputting these values into Analytical Graphics Inc.'s (AGI) space simulations software called Systems Tool Kit (STK) via a plug-in called Orbit Tuner, as shown in Fig. 2.23. This image the true anomaly was adjusted to 145 degrees to bring it over the Atlantic closer to North America for visualization.

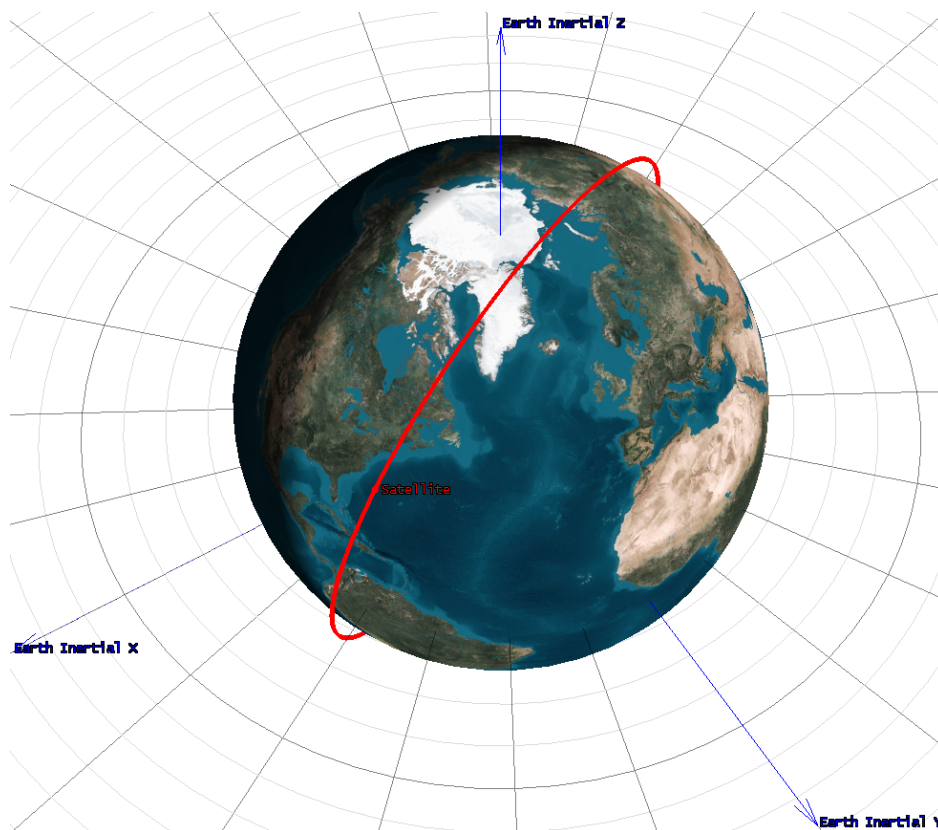
Permutations on ExampleSat's parameters would only change the altitude according to Table 2.4. In this way we now have a generalized orbit for CubeSats that will allow for direct and realistic comparisons for mission planning or analysis where we only need to adjust the altitude. We should note that as more launches occur from the ISS a second inclination of 51.6 degrees could be added to the set of ExampleSats.

Having determined the orbital mechanics for ExampleSat-165 (and ExampleSat's -480, -605, -715) we can now look to see if the population of CubeSats reveals any trends within their communications design.

### 2.5.3 CubeSat Downlink Selection

The selection of transmission output power has an impact on the overall mission design for any CubeSat. This choice impacts not only the ability to communicate successfully, but also the energy budget of the spacecraft and, as such is an important design criteria. In 2012 Arnold *et al* [Arnold2012] suggested that CubeSats typically have a power budget ranging from 2 to 8 Watts. However, as we can see in Fig. 2.24 the histogram of CubeSat transmit power is actually less than what Arnold *et al.* had suggested and is distributed about 1 W regardless of transmit frequency.

The data collected in Fig. 2.24 was tabulated based on Klofas *et al.*'s work with additional sources such as the TUB SSD [Buscher2014] and project websites for specific CubeSat systems when available. The data within Klofas *et al.*'s work is also updated semi-annually



**Fig. 2.23:** ExampleSat-165's orbit based on CubeSat community survey data.

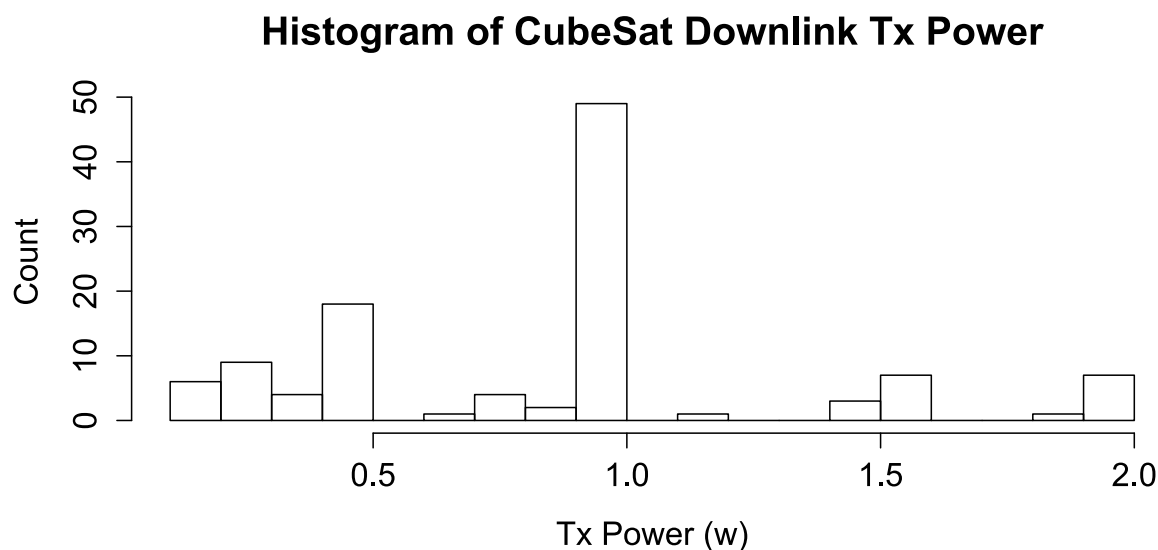
(<http://www.klofas.com/comm-table/>) with the latest available as of 14 April 2016 at the time of this writing. For consistency within any derived models the remainder of the CubeSat community survey was restricted to data from the period of 2003 to 2013.

Alongside the transmit power of ExampleSat-165's radio another important criteria is the choice of downlink frequency. Table 2.5 shows the potential downlink frequencies that are possible for ExampleSat-165. In 1995 Shtun outlined the spectrum use for Canadian operators within the Radio Amateurs of Canada Band Plans[Shtun1995].

**Table 2.5:** Satellite Downlink RF Bands in Canada[Shtun1995].

Band	$\lambda(\text{m})$	Min (MHz)	Max (MHz)	Amateur Notes
UHF	0.70	435.0	438.0	Satellites, Secondary User
VHF	2	145.8	146.0	Exclusive, ARISS
VHF	2	144.39	144.39	APRS, Primary User

Recall that that any radio transmissions from any satellite needs coordination for frequency allocation via the ITU at a global scale. The assumption that “its on the ham bands, and we are licensed hams, so we don't need additional approval” is false. It is for this reason that we must consider if radio amateurs are primary or secondary users on the band. We cannot construct an overlay network for a downlink from ExampleSat-165 if it would interfere with a primary user who isn't a radio amateur. For the sake of completeness its worth noting the



**Fig. 2.24:** Histogram of CubeSat Downlink Transmitter Power. Data derived from Klofas *et al.* [Klofas2013].

total set of applicable bands for both uplinks and downlinks while paying close attention to the large list of non-interference restrictions.

**Table 2.6:** Amateur Satcom bands in Canada [Shtun1995].

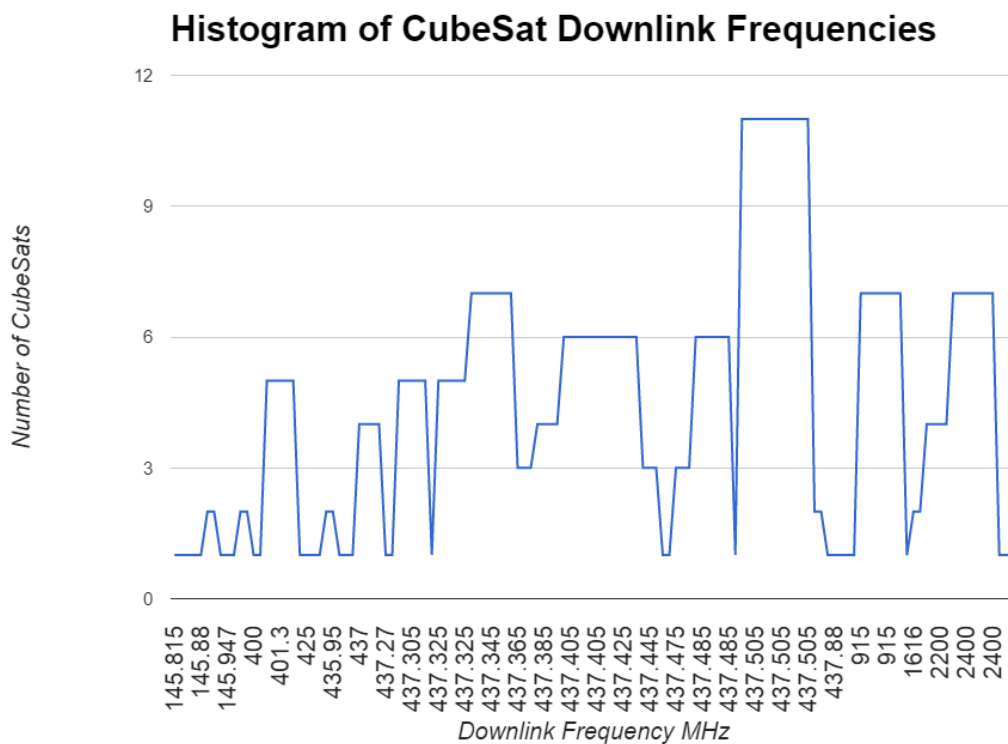
Band	$\lambda$ (m)	Min (MHz)	Max (MHz)	Amateur Status	Notes
VHF	2	144	146	Primary	Preferred: 145.8 to 146
UHF	0.7	435	438	Secondary	
UHF	0.23	1260	1270	Secondary	uplink only
UHF	0.13	2400	2410	Secondary	* 2403 to 2408
UHF	0.13	2430	2438	Secondary	* 2433 to 2438
SHF	0.09	3400	3410	Secondary	NA in ITU region 1
SHF	0.05	5650	5670	Secondary	uplink only
SHF	0.05	5830	5850	Secondary	downlink only

\*High Data Rate at the applicable frequencies.

In Table 2.6, not all of the bands are practical for ExampleSat-165's downlink use. The 1.26 GHz UHF and 5.65 GHz SHF bands are for uplinks only. Both of the 2.4 GHz to 2.438 GHz and 5.83 GHz to 5.850 GHz regions are impractical since amateur radio is a secondary user within the these crowded Industrial, Scientific and Medical (ISM) WiFi regions. 435 MHz to 438 MHz also has the non-interference status but is a popular band for amateur radio applications. Of the available frequencies only the 144 MHz to 146 MHz VHF band available for amateur radio as a primary user.

To understand the CubeSat community's design choices for communications we need to explore the most popular downlink frequencies of choice. Fig. 2.25 shown below is a histogram

of CubeSat downlink frequencies based on the same population as described by Fig. 2.24 via Klofas *et al.* and the TUB SSD.



**Fig. 2.25:** Histogram of CubeSat Downlink Frequencies in VHF and UHF. Data derived from Klofas *et al.* [Klofas2013].

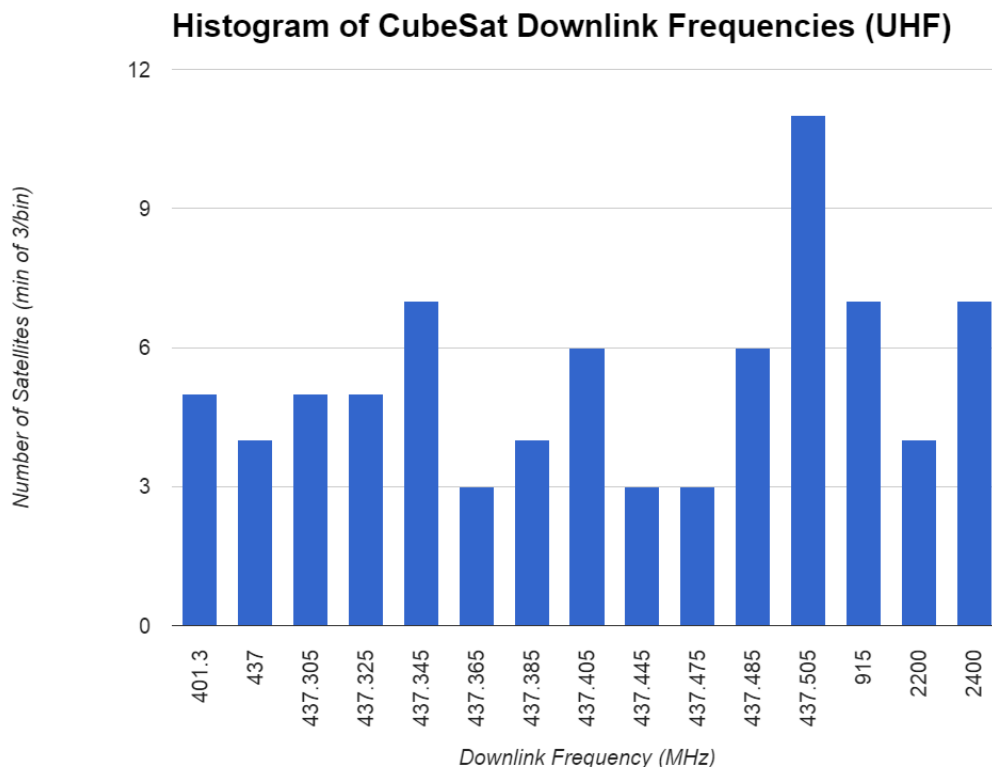
From Fig. 2.25 we can see that the distribution of frequencies used in CubeSat missions is skewed towards the UHF band for satellite downlink. This corresponds well with Table 2.7 and Table 2.6 when we consider that with a higher frequency a higher data rate modulation is available which is needed for the growing data size for CubeSat science missions. Fig. 2.26 shows a histogram of CubeSat use specifically within the UHF frequencies.

This reasoning leads us to the further exploration of the UHF downlink usage as shown in Table 2.7. This analysis gives a median frequency for any ExampleSat using UHF to be 437.385 MHz.

**Table 2.7:** Distribution of UHF downlink Satcom frequencies. After:[Klofas2013]

	Downlink Frequency (MHz)
First Quartile	437.305
Median Frequency	437.385
Third Quartile	437.485

For our further analysis we will be using the 144.39 MHz centre frequency as all of the I-Gates within North America are tuned to that frequency. In 1999 Bob Bruninga (WB4APR)



**Fig. 2.26:** Histogram of CubeSat Downlink Frequencies in UHF only. After:[Klofas2013]

proposed a common “GPS calling frequency” for APRS mobile applications of 445.925 MHz. So far, this idea has not been widely adopted. We first have to explore if ExampleSat-165 could connect to the existing network. Future systems design would likely involve a second analysis using the UHF median frequency outlined in Table 2.7.

## 2.5.4 CubeSat Downlink Modulation, Baud & Protocol

Constructing any robust models for ExampleSat-165’s communications design requires us to also look at the modulation and baud used. An in-depth network analysis of both the modulation as well as the AX.25 protocol itself was conducted by Jan-Hielke Le Roux for his thesis with Stellenbosh University in December of 2014 [LeRoux2014]. As part of his work in creating a discrete event satellite network simulation tool using Python, Le Roux conducted a similar summation of the CubeSat population, also based on Klofas *et al.*’s work, to compare AX.25’s capabilities with its Forward Error Correction variant FX.25. His objective was to devise an optimal network protocol strategy for nanosatellite communications, which he determined to be a combination of both AX.25 and FX.25. In particular he suggests FX.25 for the early sections of a satellite’s pass and then switching to AX.25 once closer through a combined decoding algorithm on a Field Programmable Gate Array chipset. His investigations also determined that FX.25 only exceeded the performance of AX.25 once the BER was above  $10^{-3}$  and the data field size above approximately 60 Bytes. For those

wanting to know specifics about AX.25 capabilities under different conditions of modulation, baud, Bit Error Rates, data field sizes and antenna gains, Le Roux’s thesis is an excellent resource [LeRoux2014].

However, our virtualized ground station network is not yet focused on an optimized nor idealized solution, but rather a connection with the existing infrastructure of APRS I-Gates. That said we must still see what the population of CubeSat’s uses to create our generalized model. Referencing Table 2.8, AX.25 is the most popular protocol while the modulation type of preference was Audio Frequency Shift Keying as described in Sec. 2.4.2.

**Table 2.8:** The population of protocol types within CubeSats. Data derived from Klofas *et al.* [Klofas2013].

Protocol Type	Population of CubeSats
AX.25	68
Proprietary	14

Interestingly, Le Roux’s thesis did not explore AFSK modulation directly. This is understandable since his interests were in an optimized communications solution and AFSK *may* be inefficient in both RF spectrum use and energy per bit required for successful reception. For an optimized design one could also explore Quadrature Amplitude Modulation (QAM) as it is popular within the commercial satellite industry particularly for Digital Video Broadcast. Additionally it is a well understood modulation technique and can be used for both analog and digital transmissions.

A review of the literature shows that there is a distinct lack of documentation for AFSK on the BER, Energy per bit relative to Noise ( $E_b/N_0$ ), modulation index, spectral efficiency or other derivations typical for digital transmission techniques. This may be in part due to how AFSK is an early bridging technology between the analog domain and the digital domain. This leads to ambiguity of how to derive any calculations for it as most the communications community is used to dealing with such calculations as applied to the RF carrier signal, rather than the audio frequency baseband signal. It also may be due to its limited use within the relatively smaller Amateur Radio community. However, as Table 2.9 shows, AFSK is the second most popular choice next to regular Frequency Shift Keying (FSK) for CubeSat communications.

**Table 2.9:** The population of modulation types within CubeSats. Data derived from Klofas *et al.* [Klofas2013].

Modulation Type	Population of CubeSats
AFSK	24
BPSK	10
FSK	32
GFSK	5
GMSK	9
MSK	4



The population of CubeSats also shows a preference for data rates of 1200 baud and 9600 baud as shown in Table 2.10. These bauds correspond to the ones commonly used for APRS on the 144 MHz and 440 MHz bands respectively.

**Table 2.10:**

Population of CubeSat data bauds. After: [Klofas2013] and [Buscher2014]

Baud	Population of CubeSats
1200	42
2400	2
4800	1
9600	32
77000	1

### 2.5.5 CubeSat Antenna Designs

Before finishing with the transmission system design for ExampleSat we will also need to know what type of antennas are used by the CubeSat population. Klofas *et al.* and TUB SSD as well as various CubeSat project pages provide additional details. Table 2.11 shows the population of CubeSat antenna designs clusters around dipole, and monopole antennas. We should note that the turnstile antenna is growing in popularity for CubeSats due to its far field antenna pattern having a smaller null area, if at all, by comparison.

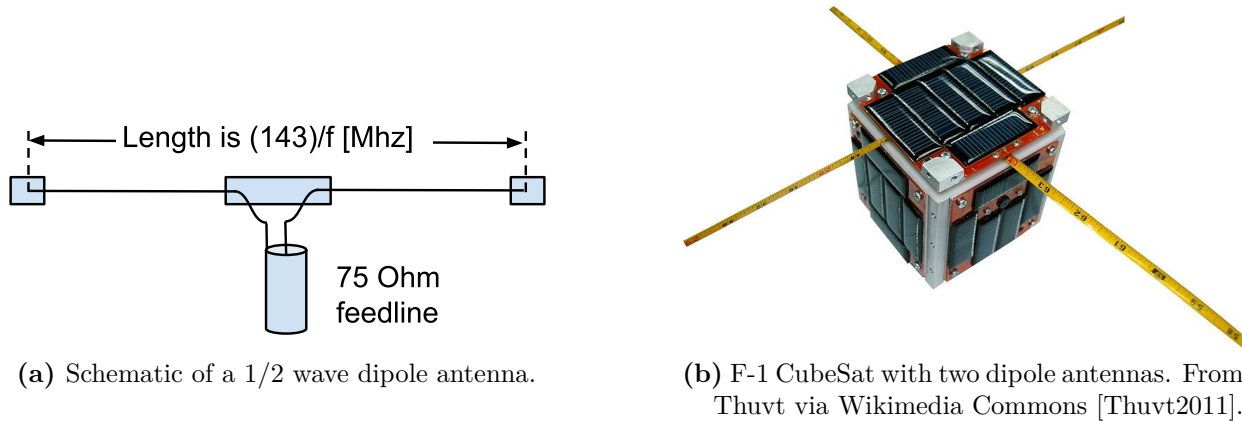
**Table 2.11:**

Population of CubeSat antenna designs. After: [Klofas2013] and [Buscher2014]

Antenna Type	Population of CubeSats
Dipole	33
Monopole	27
Turnstile	18
Patch	16
Quadrafilar Helix	1

Dipole antennas are among the most popular types of antennas due to the simplicity of their design and thus ease of deployment. Two rods or wires orientated within their ends opposite each other along the same axis are connected to the feed line from the radio. A very simplified diagram is shown in Fig. 2.27a, whereas Fig. 2.27b [Thuvt2011] shows the F-1 CubeSat with both the UHF and VHF dipole antennas deployed. Tape measure is used as the antenna material because its shape leads to it naturally unfolding and staying rigid. The fact that a build team can cut it to its electrically resonant length is as added bonus.

Returning to the 4nec2 antenna modeling software mentioned in Sec. 2.4.5 we can create models for ExampleSat-165's possible antennas, Fig. 2.28a, and the associated far field pattern, Fig. 2.28b. With models for the I-Gate ground stations antennas, ExampleSat-165's



**Fig. 2.27:** Example Dipole antennas

antenna variations, and model for the potential orbits, we can now expand ExampleSat-165s model definition.

Based on the modulation type, bauds and protocols we have seen from the CubeSat population we can now determine the transmitter for ExampleSat-165:

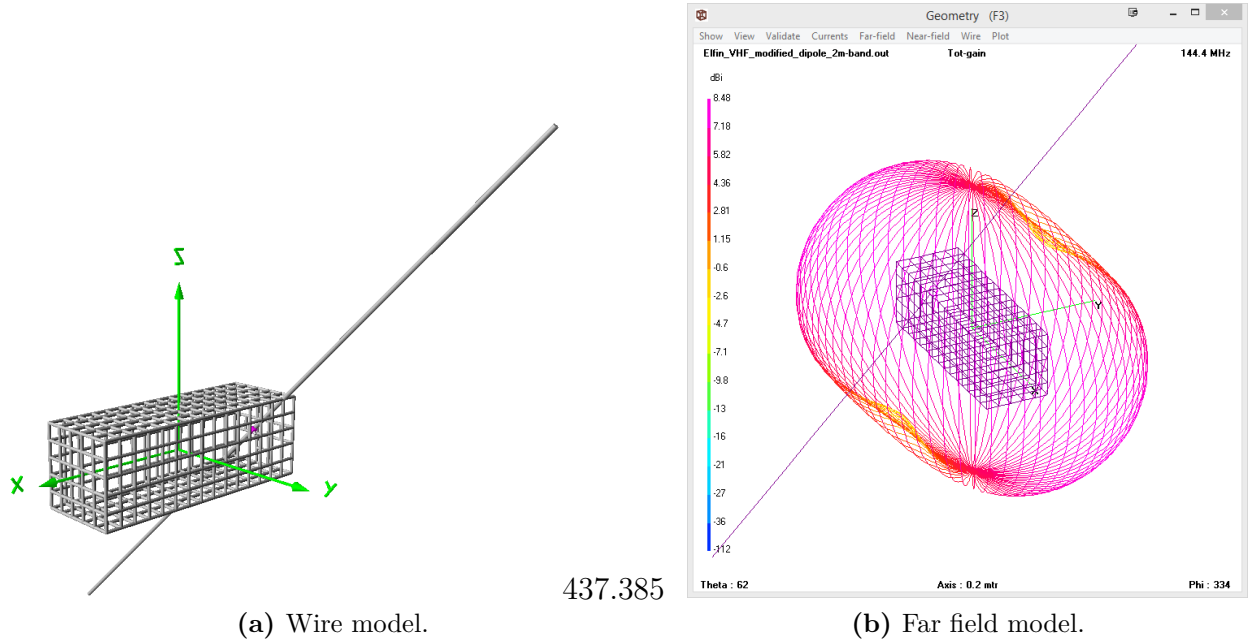
- AFSK: popular with CubeSats and as used by the terrestrial APRS network
- AX.25: popular with CubeSats and as used by the terrestrial APRS network
- 1200 baud: popular with CubeSats and as used by the terrestrial APRS network
- 144.39 MHz: popular with CubeSats and as used by the terrestrial APRS network
- 1W of transmission power: popular with CubeSats
- Dipole antenna: popular with CubeSats. Closely followed by monopoles and turnstile designs

Now that we have a baseline for models to use within any simulations, we need to establish some measure of performance so we can evaluate our system design. As discussed in Sec. 2.6, our choice impacts our ability to make comparisons in addition to which tools are used to do the evaluation.

## 2.6 Metrics of Performance

### 2.6.1 Bit Error Rate and $E_b/N_0$

$E_b/N_0$  is defined as the ratio of the signal Energy Per Bit ( $E_b$ ) to the Noise Power Spectral Density ( $N_0$ ), the combination of which are used as a measure of the how “strong” a received signal is within a digital transmission system. If  $E_b/N_0 = 1$  it implies that the signal and the noise power levels are equal. The  $E_b/N_0$  ratio is commonly used as Figure of Merit for



**Fig. 2.28:** Possible models of ExampleSat-156's antenna design in 4nec2.

digital communications as it allows for a comparison between communications systems using different modulation schemes, encoding schemes and data rates.

Lets us consider two transmitters using BFSK (actually AFSK as described in Fig.Sec. 2.4.2) in which we can define their output in time as:

$$s_1(t) = \cos(2\pi f_1 + \phi_1)$$

$$s_2(t) = \cos(2\pi f_2 + \phi_2)$$

Where  $s_1$  and  $s_2$  are the output of the transmitters as a function of time. Then  $f_1$  and  $f_2$  are the transmission frequencies and  $\phi_1$  and  $\phi_2$  are the phase components of each. If  $\phi_1 = \phi_2$ , then the two transmitters are considered to be coherent in phase. If the receiver had knowledge of these starting phases then coherent detection techniques could be applied. However, the difficulty of synchronization between the transmitters, and that AFSK predates many current digital modulation schemes tells us that coherent detection techniques are not used for APRS and thus should not be a part of our models. It may be worth considering for future systems design with regard to cognitive radio applications.

For any digital transmission we can define the Bit Error Rate (BER) as the number of bits within a data stream that have been altered by noise, interference, distortion, bit synchronization, inter-symbol interference or other factors as the information traverses the channel medium. We define the relationship between probability of any bit being in error

( $p_{BER}$ ) and  $E_b/N_0$  as [RF2013]:

$$p_{BER} = \frac{1}{2} \exp\left(-\frac{E_b}{2N_0}\right) \quad (2.5)$$

Colloquially we will refer to  $p_{BER}$ , which represents the probability of bit error rate for the non-coherent audio variant of Binary Frequency Shift Keying, simply as  $r_{BER}$  to encapsulate both the probability of a bit being in error within an AFSK non-coherently modulated data stream.

$$r_{BER} = \frac{1}{2} \exp\left(-\frac{E_b}{2N_0}\right) \quad (2.6)$$

Typically, commercial satellite communications systems use digital modulation schemes such as variants on Phase-Shift Keying, Frequency-Shift Keying, Amplitude-Shift Keying or Quadrature Amplitude Modulation to convey information upon the carrier signal. These different forms of modulation each have different theoretical  $r_{BER}$  vs  $E_b/N_0$  but our interest is in BFSK and thus its particular variant of AFSK for the APRS I-Gate network.

Another common measure of performance for digital communications is the Packet Error Rate (PER) which represents the number of error packets after Forward Error Correction (FEC) divided by the total number of received packets. PER is for the entire packet set and thus studies the protocol as a whole, whereas BER is for the bit-stream and looks at the individual received bit after FEC. For a normal telecommunications link design both PER and BER account for the receiver's sensitivity, which is in practical terms directly relates to the receiver's ability to discern  $E_b$  from  $N_0$ . In the case of our virtualized network of heterogeneous ground stations it would be impractical to determine or even guess each individual station's receiver sensitivity. For this reason it is more appropriate to evaluate ExampleSat-165's downlink directly by its  $E_b/N_0$  first and then look at BER to determine the packet or data payload reception.

Work done by Ralph Wallio (WØRPK) [Wallio2015], better illustrates the practical/realized relationship of  $r_{BER}$  and  $E_b/N_0$  specifically for the AX.25 protocol including packet size. WØRPK credits the efforts of Steve Goode (K9NG) for that author's work titled "BER Performance of TAPR TNC Modem, ARRL QEX, August 1983" [Goode1984] in which extensive manual BER testing was performed with a TAPR TNC modem. For completeness their relevant contributions are in Table 2.12.

As a result of both Eq. 2.5 and the evidence shown in Table 2.12 we see as expected that a stronger signal compared to the noise lowers the risk of receiving bits in error.

To add perspective via comparison with a network of similar scale we can look at Wireless Metropolitan Area Networks (Wireless MAN) in the IEEE 802.16 standard, specifically the 2009 Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems [IEEE2009], since superseded by new revisions. Our reasoning for looking at an outdated standard is that it provides us with a comparative context for APRS which in effect acts similar to a WirelessMAN but with amateurs using optionally mobile nodes. The 802.16 2009 standard specified "The BER measured after FEC shall be less than  $10^{-6}$ " [IEEE2009]. Its telling that the superseding standard does not specify

**Table 2.12:**

$r_{BER}$  to  $E_b/N_0$  for APRS. Data derived from Wallio and Goode[Wallio2015, Goode1984].

$r_{BER}$	$E_b/N_0(\text{dB})$
$1.6 \times 10^{-5}$	24.0
$1.6 \times 10^{-4}$	22.5
$1.6 \times 10^{-3}$	19.5
$1.6 \times 10^{-2}$	17.5
$1.6 \times 10^{-1}$	15.5

a BER as this would in turn mandate the receiver sensitivity which would be impractical for the creation of a robust Wireless Man industry to create a selection of products. What we can take from this superseded standard is that a BER of  $10^{-6}$  is the minimum BER for a higher data rate communications (typically between 1 to 5 Mb/s for non-line of sight instances of several kilometers [Frenzel2015]). By contrast [LeRoux2014] focused his thesis efforts on bit error performance calculations with a fixed BER of  $10^{-4}$ . For our purposes we will have to tabulate the entire set of received  $E_b/N_0$  for each ground station from ExampleSat-165 to see if any trends emerge, then determine the applicable BER. This will enable us to plot BER vs  $E_b/N_0$  and evaluate the link performance that includes orbital mechanics as a component.

In a lab environment the true BER is not a practical value. One would measure enough data to have a confidence level such that the BER was lower than a specified level. The independent timing noise and clock signals testing company called Jitter Labs uses a BER Confidence-Level calculation [JitterLabs2015]. This is used to determine the number of bits that need to be transmitted to determine if the BER at a given confidence level. Collected from Jitter Labs website BER Confidence-level Calculator .

$$l_{conf} = 1 - e^{-N[r_{BER,req}]} \times \sum_{k=0}^{n_{errors}} \frac{(N \times r_{BER,req})^k}{k!} \quad (2.7)$$

Where  $N$  is the number of repeatedly transmitted bits,  $n_{error}$  is the number of errors detected and  $r_{BER,req}$  is the specified BER that the bit stream should achieve or better to at the Confidence Level  $l_{conf}$  based on a Poisson distribution. This gives us the confidence level that the measured BER is less than the specified  $r_{BER,req}$ . Now that we have an appreciation for the modulation, frequencies involved and the measures of performance we can begin to outline some criteria for successful links. A useful reference for understanding the basic principles of data rates as they apply to satellite telecommunications is Minoli's 2009 book "Satellite systems engineering in an IPv6 environment" [Minoli2009]. Although his book is geared towards higher frequency satellite communications above one gigahertz (1 GHz), as is typical for Digital Video Broadcast and satellite internet, the principles and formulas are presented in an easy to read and practical format. Much of the following link budget formulas was influenced by this book.

To communicate with the ground stations via APRS ExampleSat-165's transmitter could be set to either 1200 baud or 9600 baud which we will refer to as  $r_{baud}$ . In the case of our

AFSK variant of BFSK is equivalent to 1.2 and 9.6 Kbps as described in Sec. 2.4.2.

$$r_{baud} = \{1.2, 9.6\} \quad (2.8)$$

From here the data rate can be transformed to represent a energy density per Hertz as a Desired Data Rate ( $dBHz$ ):

$$[dBHz] = 10\log_{10}(r_{baud}) \quad (2.9)$$

Although such a metric shown in Eq. 2.9 is not used directly it may be informative when determining the effects of Doppler shift on link quality.

Minoli [Minoli2009] defines the  $E_b/N_0$  as:

$$\frac{E_b}{N_0} = \frac{p_{rx}}{N_0} - 10\log_{10}(r_{baud}) \quad (2.10)$$

Where  $p_{rx}$  is the received power in dB at the ground station. From here we can see what the BER of the potential output data rate will be. Recall that [RF2013] states that the Probability of Error within a digital transmission as:

$$r_{BER} = \frac{1}{2}erfc\left(-\frac{E_b}{2N_0}\right) \quad (2.11)$$

Note that in this calculation we assume a spectral efficiency of 1.0 bps/Hz of Bandwidth.  $erfc$  is the complimentary error function.

Now we can accept or reject the output combination if the following is true:

$$r_{BER} \leq r_{BER,req} \quad (2.12)$$

In an additive white Gaussian noise environment the Shannon Capacity ( $c_{awgn}$ ) of a channel is:

$$c_{awgn} = w \times \log_2\left(1 + \frac{p_{rx}}{N_0 w}\right) \quad (2.13)$$

In this case  $p_{rx}$  over  $N_0 w$  is actually the received signal to noise ratio  $p_{rx}/N_0$  calculated earlier with  $w$  as the available bandwidth in Hz from Table A.2 which describes the possible channel width the frequency bands available for amateur radios satellite communications. As such equation Eq. 2.13 becomes:

$$C_{awgn} = w \times \log_2\left(1 + \frac{p_{rx}}{N_0}\right) \quad (2.14)$$

To summarize, for the combination of variables of each iteration of the link to be considered a successful link design the following three conditions must be true:

$$r_{BER} \leq r_{BER,req}$$

$$c_{awgn} \geq r_{baud} \quad (2.15)$$

As a general rule we could call a link successful if the received  $E_b/N_0$  is at least twice that of the minimum required,  $(E_b/N_0)_{req}$ . Thus we can also state the following:

$$\frac{E_b}{N_0} - \left( \frac{E_b}{N_0} \right)_{Req} \geq 3[dB] \quad (2.16)$$

### 2.6.2 Stationarity of $E_b/N_0$

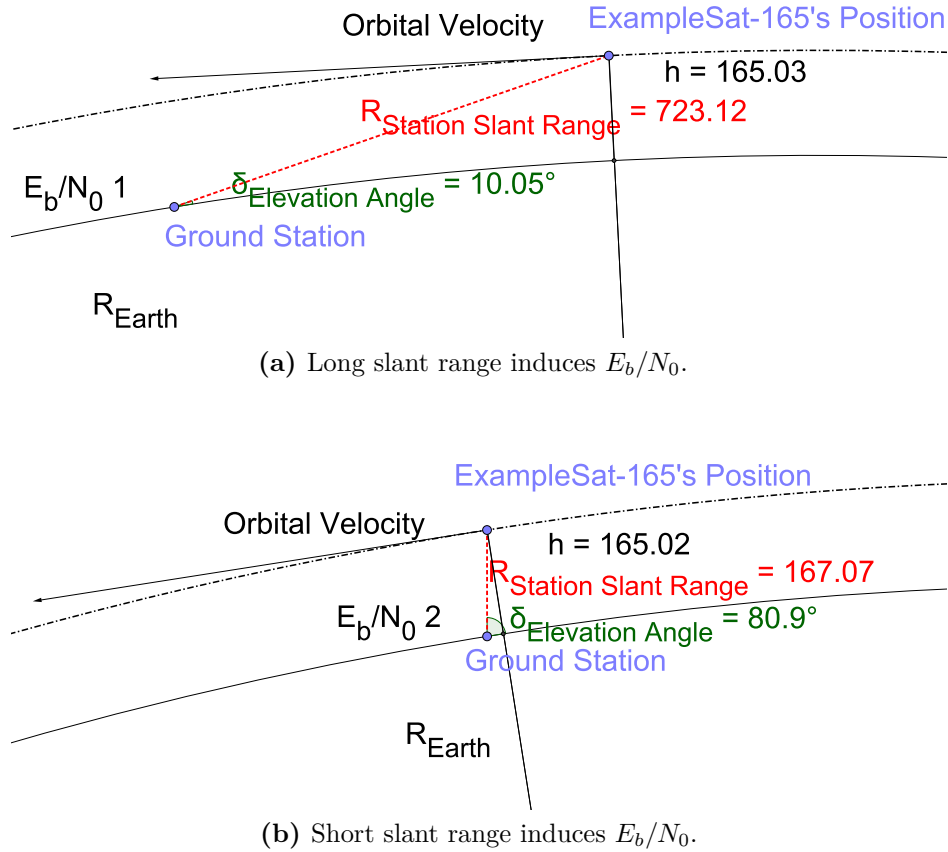
Let us revisit the topic of slant range as discussed in Sec. 2.5.1 which illustrates the impact on spreading loss on the received signal from ExampleSat-165. Fig. 2.29 is a close example of how the slant range from ExampleSat-165 to the ground station changes. Note that the images in Fig. 2.29 are to scale and the units for distance are in kilometers.

The slant range between the two Sub-figures should also demonstrate a difference in the received energy per bit ( $E_b$ ) due to the spreading loss as the satellite approaches and then recedes from the ground station. This implies that the received  $E_b/N_0$  to a single ground station within our proposed virtualized network will vary with time.

Any time series data can be called stationary if its first and second moments as well as the autocorrelation function of the series do not change with time. In many instances this would imply the the mean and variance remain stable with time and that we notice no trends across time with the data set. Stationarity also precludes fluctuations over time.

A Strict Sense Stationary process is a stochastic process whose joint probability distribution does not change when shifted in time. The statistical distribution of any collection of time series variates never depends on time. In looking at the mean, variance and any other higher order moment of any variate is the same, whichever variant is chosen across the time series. The University of Oxford, Department of Statistics, Time Series Analysis Section III, Michaelmas Term 2010 Notes [#Statistics2010] states “A stationary series has a well-defined mean around which it can fluctuate with constant finite variance”. In 1991 Challis and Kiney stated that Stationarity is a defined quality of a process in which the statistical parameters (mean and standard deviation) of the process do not change with time [#Challis1991]. One indication of a stationary process is that its autocorrelation function varies with lag only and not with the time at which the function was calculated.

A classic example of a stationary signal is Gaussian White Noise as show in Fig. 2.30a. The plot shows that although the time series data has differing values at differing points the mean remains the same across time. One may be thinking, but if we calculate the mean value across the data series it summates to single value and thus does not actually exist across time as a set of values.



**Fig. 2.29:** Changes in slant range induce non-stationarity of revired  $E_b/N_0$  over time.

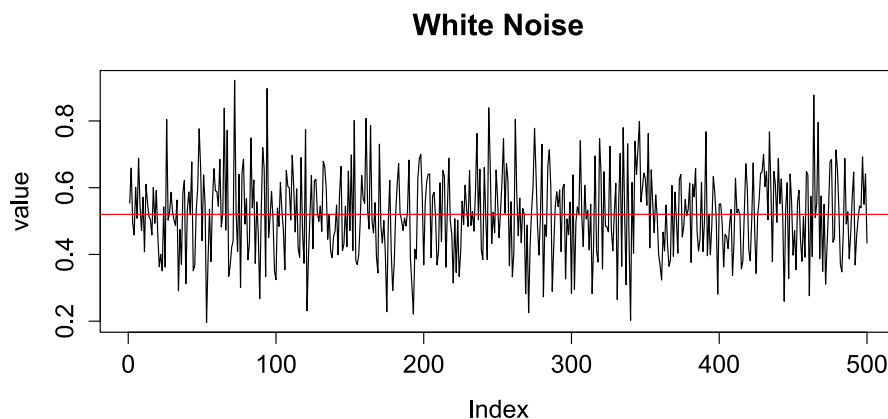
In this case an illustration of a series that does change with time may provide clarification. Fig. 2.30b demonstrates Brown Noise with the same number of data points generated using the same power law function where  $1/f^0$  creates White Noise and  $1/f^2$  creates Brown Noise. The R programming language [Ihaka2016] was used to create this data series and perform analysis on them. R Studio [Allaire2016] was used as an Integrated Development Environment (IDE) for its graphical user interface (GUI) along with the TK95 function from the “{RobPer}” package [Thieler2016]. R is discussed in greater detail in Sec. 3.3.

Using the mean as the first moment is a reasonable fit as a measure of central tendency for the distribution of White Noise in Fig. 2.30a. We can visually see that the mean would not be a suitable measure of central tendency for the distribution of values within Brown Noise.

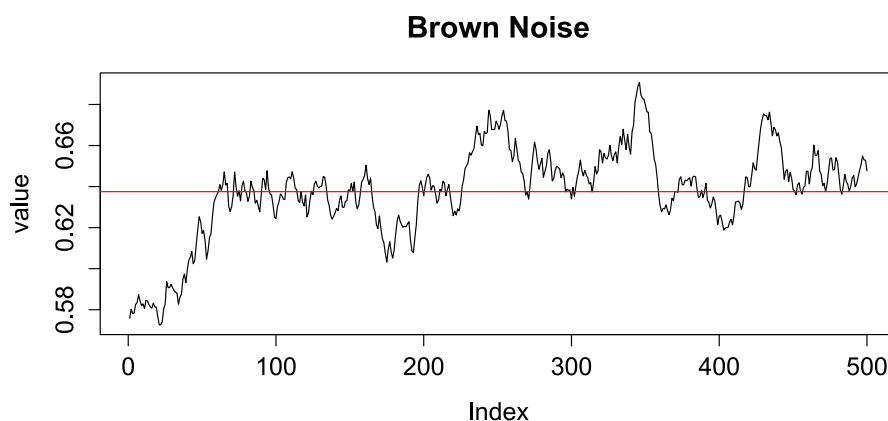
For example, Brown Noise can be used to describe the random motion of gas particles due to collisions with atoms or molecules. The value for the next position of a gas particle is based on its remaining momentum from the last collision. Its motion is random but biased due to this remaining momentum. Looking again at Fig. 2.30b for Brown Noise, we can see that subsequent values appear dependent on the current value. The total variance for the set of 500 points may be similar between White and Brown Noise, but the variance between individual points within Brown Noise is much smaller.

Strict Sense (or Strong Sense or First Order) Stationarity is rarely found in nature and thus





(a) White Noise time series data.



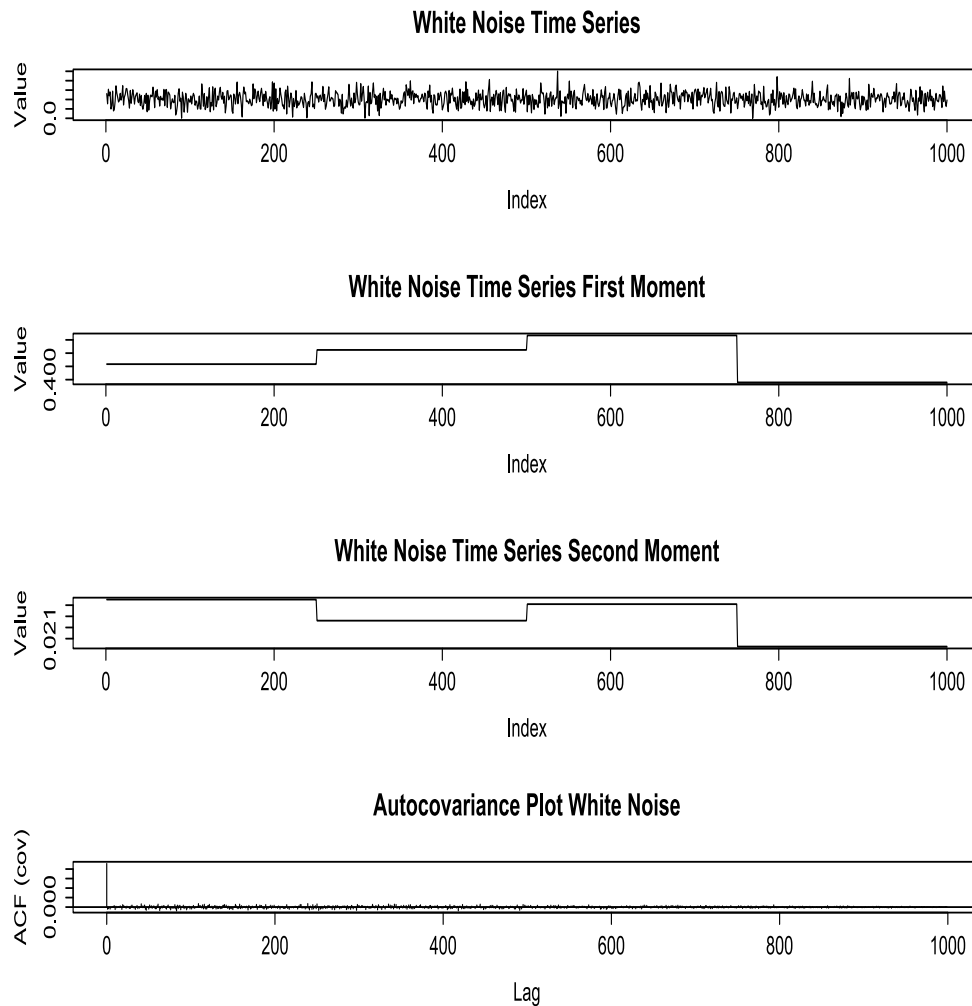
(b) Brown Noise time series data.

**Fig. 2.30:** White and Brown Noise time series data generated via power law.

Weak Sense (or Second Order) Stationarity was coined such that it only requires the the first moment, in many cases the mean, and the autocorrelation do not vary with respect to time. The variance is included within the autocorrelation and would also be constant across time as would the autocovariance. However, higher order moments such as skew, kurtosis, covariance, coskewness, cokurtosis could vary with time and still meet the definition of Weak Sense Stationarity. Fig. 2.31 shows a plot of White Noise, and its first and second moments across windows of 250 samples. Notice that the autocovariance is flat across time. Compare this with the autocovariance shown in Fig. 2.32 where the autocovariance is drastically different across its lag in time.

In most cases we are interested in the stationarity of a signal or process over a finite period of time. We rarely investigate the entirety of a time series at once. Thus we should modify the terminology to Finite Window Weak Sense Stationarity (FWWSS) to indicate that we are looking at a sliding window of the time series data.

From an analysis standpoint non-stationary signals create problems because a number of

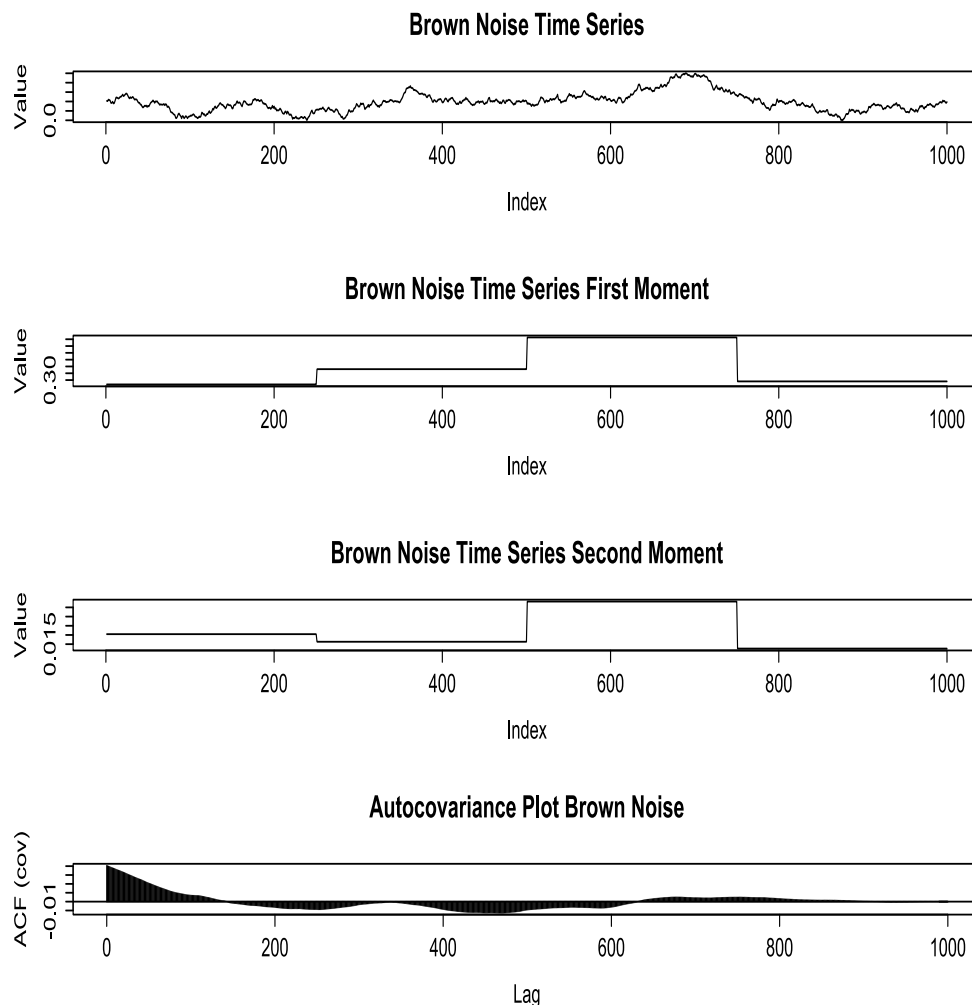


**Fig. 2.31:** Plots of White Noise moments and Autocovariance.

tools traditionally applied for both signals analysis and statistical analysis are no longer applicable. Fourier transforms require that the signal in question is from a linear time-invariant system. Brown noise is not time-invariant and thus a Fourier transform would not provide useful insight such as a spectrograph or distribution of frequencies. Furthermore classic statistical tools such as z-score, Student T-tests, and ANOVA become unreliable. ANOVA is based on the ratio of Fisher F-distributions which is sensitive to the normality of the underlying series.

In a typical time series analysis one would first create a run sequence plot to visually determine if any seasonality is present in the data and also look for generalized trending. One would then either detrend the data or remove the seasonality by differencing. If the data contained a trend, curve fitting could be done and residuals could then be plotted to determine its fitness. This fit curve would represent the system of interest or entire time series.

Contrary to typical time series analysis, we are not initially interested in determining the



**Fig. 2.32:** Plots of Brown Noise moments and Autocovariance.

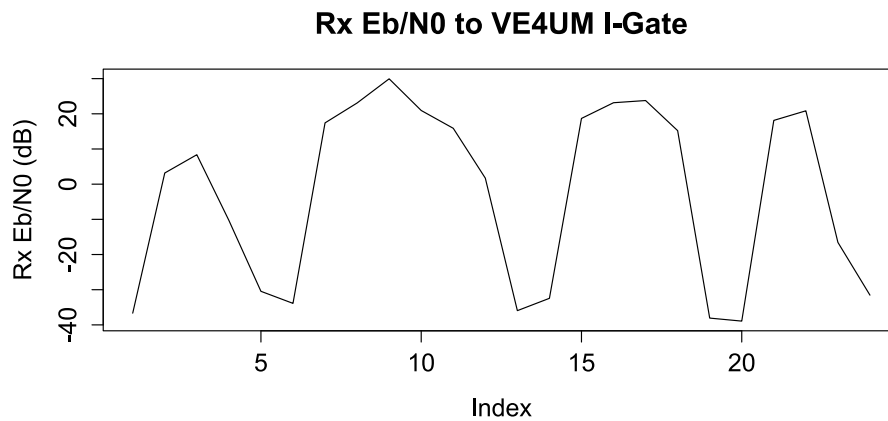
stationarity of the entire series. For example, in econometrics the typical process is to visualize the time series, to stationarize the time series, to plot the Autocorrelation Function (ACF) and Partial Autocorrelation Functions (PACF) of the series, to optimize the parameters of the ACF and PACF of the series, and then to build an Autoregressive Integrated Moving Average (ARIMA) model of to make predictions. These steps were described by Tavish Srivastava's Analytics Vidhya article in 2015 [Srivastav2015]. However, we are not interested in modeling the entire time series as a complete set at once.

We will define our received sample duration for the received  $E_b/N_0$  occurs over a twenty four hour (24h) period. This represents an operationally relevant period of time as several passes should occur for any spacecraft below a Medium Earth Orbit altitude of approximately 20,000 kilometers. We know that the azimuth and elevation angles from each I-Gate ground station are different based on the true anomaly of ExampleSat-165 within its orbit.

Each I-Gate will see ExampleSat-165 at a different angle to the the geometry from that point on the ground. Based on these facts we can infer that our time series may appear to

be non-stationary due to its duration and input variables. If we alter the orbital altitude of our satellite, the entire set of outputs would change. The possible number of input variables limits our ability to conceptualize a generalized model that combines all the possible orbital elements with the telecommunications factors. Thus, any ARIMA model of received  $E_b/N_0$  could only be for a selected set of orbital elements for a limited period of time based around a specified epoch.

We can see that Fig. 2.33 shows the simulated received  $E_b/N_0$  from ExampleSat-165 to an I-Gate located at the University of Manitoba over a 24 hour period. Note that the portions of time in which the spacecraft is not within Line of Sight (LOS) of the ground station are removed. Note that in this image Index is unitless and simply indicates the order of occurrence for each variate.



**Fig. 2.33:** Non-stationarity of the received  $E_b/N_0$  for ExampleSat to a single ground station. Note that Index is unitless.

The time series data for  $E_b/N_0$  varies with time, and it is not entirely cyclical. The series may be a stationary ergodic process over a much longer period of time. Determining this requires that we know at which time the spacecraft will pass over the same sub-satellite point on the Earth with the same orientation relative to the surface, many days into the future. This would also assume that orbital perturbations were ignored or negligible.

We need to find a generalized method of determining a window size in which the time series data exhibits stationarity for a statistically useful portion of the time series as a whole. In this process the stationarity need not be contiguous throughout the entire time series. Our goal is to see if by using a virtualized network of ground stations, effectively adding a large number of receivers, that the combined set of receivers enables a stationary signal over a window period. If this is the case a larger set of signal processing and statistical tools will be available to further enable more advanced communications methods.

An important feature of a stationary time series is that its autocorrelation function depends only upon its lag and does not change with the time at which the function was calculated. Stationarity can be considered a relative term rather than an absolute since few observable systems can truly possess the characteristics required for strong sense stationarity. At best one can only sample a process for a prolonged period of time compared to the lowest frequency

component in the data. If we collect data for a short period of time compared to the wavelength of the data we would believe that a non-stationary process is stationary.

In 2002, Marice R. Masliah [Masliah2002] stated that based on his literature review there are two approaches for looking at stationarity; parametric and nonparametric. He further elaborates that parametric approaches are typically used when investigating within the time domain, whereas non-parametric methods are used within the frequency domain. Nonparametric tests are not based on *a priori* assumptions that the population under test is normally distributed [Betha1991]. As a byproduct, nonparametric tests are less statistically powerful (able to detect a change when the change truly happened) than their counterparts. This leads to non-parametric tests requiring more data (between 5% to 35% more) to accomplish a similar statistical power. Fortunately for the time series data represented by the received  $E_b/N_0$  at our I-Gate ground stations we have ample data points to work with from a 24 hour sampling period. Thus a non-parametric test such as the Phillips-Peron test should suffice.

The Phillips-Perron (PP test) method is a unit root test of univariate time series. A linear stochastic process will have a unit root if 1 is a root of the characteristic equation for the process. If so, such a process is defined as non-stationary. The PP test was developed by econometrics researchers Peter C. B. Phillips and Pierre Perron in 1988 within their paper “Testing for a unit root in time series regression” for the journal *Biometrika* [Phillips1988]. As a testament to the utility of their method the paper has been cited 12884 times as of July 2016, largely within the field of econometrics.

Arguably, the field of econometrics has made greater strides within time domain analysis than much of electrical and or computer engineering. Econometrics as a field is intensely focused on studying time series data as it relates to commodities prices and predictions. Researchers in this field spend time studying the fundamentals of statistics and are forced to develop robust and reliable techniques. To quote from Phillips and Perron themselves: “*Our approach is nonparametric with respect to nuisance parameters and thereby allows for a very wide class of weakly depended and possibly heterogeneously distributed data*”. The PP test is based on estimating the first-order autoregression with a constant and may include a time trend. The method then calculates the classic z statistic based on this transformed autoregressive trend. However, in Chapter 5 we will be using more classic methods related to the first order moment to compare if the PP test is a useful measure of stationarity for our time series data.

### 2.6.2.1 The importance of stationarity

We can think of stationarity as a way of modelling the dependence of a variable on the structure of a system. Variables can be dependent upon each other in many different ways, but can be independent in only one. If a systems exhibits stationarity we can state that one or more variables of that system are independent, thus we can then model other variables from that system around the independent variable. If we can establish the stationarity of our received signal from a satellite, we can uncouple one or more of its variables from time, and thus properly begin to model the properties of the system. Independence of a variable allows us to apply statistical tools such as the law of large numbers or the central limit theorem for example. Without stationarity many assumptions about a system may not hold true.

When studying a time series statistically, we are often concerned with the expected value

of a variable, its variance, and the correlation between values several periods apart of a set of values. The expected value for a time series is much easier to understand without a codependency upon time. Independence of a variable requires stationarity. Furthermore, systems analysis methods based on Laplace techniques such as Fourier transforms require a time-invariant system. If a system (or signal in our case) is considered stationarity, and thus independent, then further methods such as finite/infinite impulse response, state space controls and frequency response controls can be applied.

In Sec. 6.3.2 we will explore the possibilities for Cognitive Radio (CR) in which a radio system is capable of analyzing the channel environment, making predictions about it, and then acting based on those predictions. In order to make predictions the channel environment must be well understood and predictable in the first place. Predictions based on sampling across time for a channel are easier to accomplish in a stationary system, rather than one in which the variables are interdependent upon each other as well as the time of measurement.

As an example, if a time series is consistently increasing over time, any sample mean and variance will grow along with the sample size. Such values would underestimate the mean and variance of future samples. As a result, predictions from regression within a non-stationarity time series could be dubious. Stationarity allows us to treat the system as predictable and thus our common suite of both statistical and signal process methods could be applied to the time series. For this reason, we want to examine the stationarity of the received signal to the virtual ground station network, to see if in future we can apply such techniques.

## 2.7 Doppler Shift

One important factor for any satellite link analysis is how to accommodate Doppler shift. If a spacecraft is in LEO it could be traveling at up to 8 km/s relative to fixed point of the ground. In the classic ground station design as shown in Fig. 2.1, the satellite would downlink on a fixed frequency while the ground station would adjust its receiver's frequency to accommodate. One key problem with the virtualization of ExampleSat-165's ground stations is that we would not have any direct access nor control over the set of individual nodes. This implies that all of the nodes in North America are tuned to 144.39 MHz to participate in the terrestrial network and do not adjust their receiver frequency for ExampleSat-165's downlinks.

One option would be to adjust ExampleSat-165's transmitter frequency to accommodate the Doppler shift. The OUFTI-1 ( launched on 25 April 2016 ) CubeSat from Université de Liège [AMSATUK2013, Pisane2013] used this technique with their D-Star compatible transmitter. D-Star is one of several competing standards of digital communications available within the amateur radio community. OUFTI-1's design calls for a 145.950 MHz FSK AX.25 downlink that adjusts its transmitter to accommodate the Doppler shift when over selected regions of the Earth. This requires the spacecraft either to have a robust knowledge of its location above the Earth's surface, or the uplinking station can predict when the the CubeSat is overhead and initialize a handshake that tells the satellite what the Doppler schema will be.

This design is not practical for virtualizing the ground stations which this thesis is interested in. By tuning to a specific Doppler shifted frequency the satellite is targeting a region

of the Earth for its downlink. For maximum data downlink time and higher flexibility of locations for downlink across North America, ExampleSat-165's transmitter should remain at 144.39 MHz. However, this could introduce Doppler induced error into the bitstream if the transmission frequency with Doppler drifts beyond the channel selectivity of an I-Gate's receiver. As ExampleSat-165 passes overhead I-Gates would experience a particular Doppler shift, in Hz per second, depending on the slant range to the satellite. The maximum rate of change of Doppler shift would occur as the satellite passes directly overhead.

In 1998 Ifran Ali *et al.* described the normalized Doppler shift ( $\Delta f/f$ ) in their paper titled *Doppler Characterization for LEO Satellites* [Ali1998a]:

$$\frac{\Delta f}{f} = -\frac{1}{c} \frac{R_E r \sin(\psi(t) - \psi(t_0)) \cos\left(\cos^{-1}\left(\frac{R_E}{r_s} \cos\theta_{max}\right) - \theta_{max}\right) \omega_F(t)}{\sqrt{R_E^2 + r^2 - 2R_E r \cos(\psi(t) - \psi(t_0)) \cos\left(\cos^{-1}\left(\frac{R_E}{r_s} \cos\theta_{max}\right) - \theta_{max}\right)}} \quad (2.17)$$

Where  $f$  is the transmission frequency.  $\omega_F(t)$  is the angular velocity of the satellite in the Earth Centric Fixed (ECF) frame of reference. That is, the relative angular velocity as seen by a ground station on the Earth's surface. According to Ifran Ali *et al.* for an altitude of 500 km the mean variation of the satellite's velocity relative the the ECF is only 0.168%. Interestingly this variation decreases with lower altitude.

We can approximate the angular velocity  $\omega_F$  as:

$$\omega_F \approx \omega_s - \omega_E \cos(i) \quad (2.18)$$

Where:

$\omega_E$  is the angular velocity of the Earth ( $7.2921159 \times 10^{-5}$  rad/sec or  $4.1780746479 \times 10^{-3}$  deg./sec),  $\omega_s$  is the angular velocity of the satellite in the Earth Centric Inertial (ECI) frame, and  $i$  is the inclination of the orbit.

Returning to equation Eq. 2.17 we continue to define its variables:

$\theta_{max}$  is the maximum elevation angle from the ground station looking towards the satellite. In this instance we must assume this to be 90 degrees and that the ground stations would have near isotropic antennas and thus "see" directly overhead. This is impractical in reality but gives us a complete set of data to work from. Ideally we would need to have knowledge of the complete set of antennas for all of the I-Gate stations in North America.

$R_E$  is the radius of the Earth at 6,378.14 km.

$r_s$  is the radius from the centre of the Earth to the altitude of the satellite's orbit.

$h$  is the mean altitude of the satellite orbit.

$$r_s = R_E + h \quad (2.19)$$

$\psi(t) - \psi(t_0)$  is the angular distance between the ground station on the Earth's surface and the sub-satellite point. The sub-satellite point is a location on the Earth's surface if a line were drawn from the spacecraft's centre of mass, perpendicular to the instantaneous position of the satellite along its orbital path.

We use an eccentricity of approximately zero (0) which corresponds with data from the population of CubeSats in Fig. 2.19.

In 2000 Tabaković wrote a paper on the “Doppler Effect in non-GSO Satellite Propagation [Tabakovic2000]” in which he determined the rate of change in slant range from a ground station to a satellite’s position in orbit.

$$\frac{dD}{dt} = \frac{R_E r_s \sin \left[ \varepsilon_m + \arcsin \left( \frac{R_E}{R_E + r_s} \cos \varepsilon_m \right) \right] \sin [(\omega_s - \omega_E \cos i) t] (\omega_s - \omega_E \cos i)}{\sqrt{R_E^2 + r_s^2 - 2R_E r_s \sin \left[ \varepsilon_m + \arcsin \left( \frac{R_E}{R_E + r_s} \cos \varepsilon_m \right) \right] \cos [(\omega_s - \omega_E \cos i) t]}} \quad (2.20)$$

$$\frac{f_d}{f_c} = - \left( \frac{dD}{dt} \right) / c \quad (2.21)$$

Where:

$dD/dt$  is the rate of change of Slant Range from the ground station to the satellite in its orbit.

$\varepsilon_m$  (degrees) is the , which is the maximum elevation angle possible for the ground station antenna system.

$i$  (degrees) is the orbital inclination of the satellite.

$c$  is the speed of light in a vacuum.

$f_d/f_c$  is the ratio of frequency change in Hz/s relative to centre frequency of the satellite’s transmitter. This is the Doppler Rate as viewed from the ground station at that elevation angle at that moment in time.

The assumption in Eq. 2.20 is that we are using a spherical Earth and that the orbit’s eccentricity is nearly circular. This is done to maintain generality of the solution. Further studies should be done to determine the error between this spherical Earth Doppler calculations and the WGS84 oblate sphere model of Earth. The requirement for nearly zero eccentricity corresponds well with what we’ve seen from the population of CubeSats in Fig. 2.19

$G$  is the universal gravitational constant and  $M$  is the mass of the Earth.  $GM = 3.986005 \times 10^{14} [m^3/s^2]$

$\omega_s$  is a derived value for the angular velocity of the satellite along its orbital path to be placed back into Eq. 2.17.

$$\omega_s = \sqrt{\frac{GM}{r_s^3}} \quad (2.22)$$

The key advantage of both the Eq. 2.17 and Eq. 2.20 formulas is that they allow us to look at the Doppler shift and the change in slant range and thus the Doppler change rate by using the common orbital elements described in Sec. 2.5.1.1 as well as not requiring a specific location on the Earth’s surface. This allows for a generalized look at not only the potential problem of Doppler shift for our virtualized network but also the slant range which is the largest source of signal loss for any satellite downlink.

### 2.7.1 Doppler induced bit errors within I-Gates

One of the open questions within this research topic is if ExampleSat-165’s orbital motion induced Doppler Shift will cause the I-Gate ground stations to receive packets in error. The



fundamental design difference between a typical AMSAT/OSCAR ground station and an APRS I-Gate is that the former has two key factors when considering satellite downlinks. The first is that the antenna itself is steerable because the physical antenna as has rotor attached that is capable of adjusting its azimuth and elevation. The second is that the AMSAT/OSCAR station would have *a priori* knowledge of ExampleSat-165's orbital path and would adjust its tuned centre frequency to accommodate for the Doppler shift. We must keep in mind that all of the RF capable nodes within the underlying APRS network are independent in that they are not created, deployed or controlled in any way by a centralized authority. They do behave within the parameters of the APRS "standard" and within North America are tuned to a fixed frequency of 144.39 MHz. As operators attempting to use the APRS network to create a virtualized ground station network we have no direct control over the RF tuning of any of the nodes. The I-Gates we are attempting to use will remain tuned to 144.39 MHz and are ignorant of our downlink needs, including the desire to compensate for Doppler shift.

The nature of the Doppler shift was characterized in Sec. 2.7, but to understand its impact we must describe how this potentially impacts a receiving I-Gate. Based on the discussion in Sec. 2.4.2 we know that ExampleSat-165 is transmitting an AFSK modulated signal that uses the baseband audio to convey its bits upon the channel. By creating an example scenario using STK we can examine the expected Doppler rate (the rate of change of Doppler shift with units of Hz/s ) associated with ExampleSat-165's movement relative to VE4UM using the 2mGP ground plane antenna model as shown in Fig. 2.15.

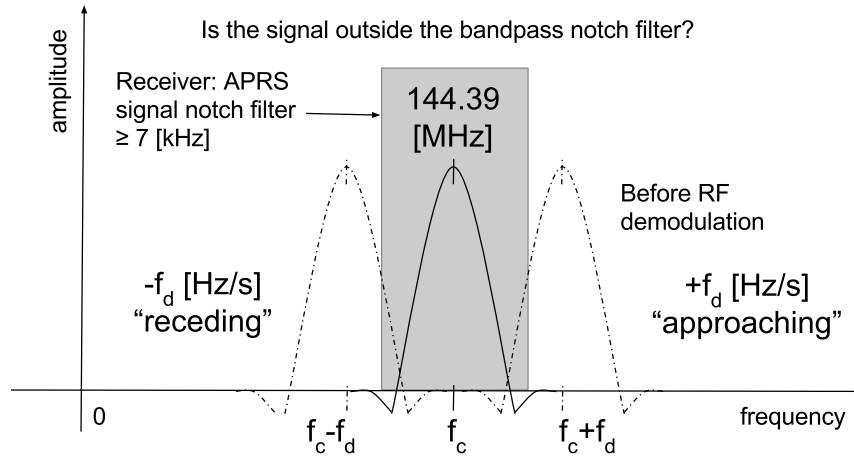
**Table 2.13:** Single ground station simulation results of induced Doppler Rate.

Measure	Doppler Rate [Hz/s]
Maximum	-90.563655
Minimum	-0.559804
Mean	-19.510081

The data in Table 2.13 tells us that the downlink signal will experience a maximum Doppler shift of -90.56 Hz each second relative to the position of VE4UM. Realistically this would be an absolute value as ExampleSat-165 first closes towards (approaches) VE4UM and then opens away (recedes) from it. But how does this impact the I-Gates ability to demodulate the received signal? The core of the issue is exemplified in several diagrams across the two categories of RF domain and audio domain. We see in Fig. 2.34 that the Doppler shift is not likely to impact the ability of the I-Gate's receiver to detect the downlink signal as being within the appropriate channel. The maximum change in Doppler rate in Hz/s may not have an impact on the receiver's kHz wide RF domain notch filter.  $f_d$  represents the Doppler rate at that instance of time.

However if we look at the audio domain the situation could be different. We see in Fig. 2.35a how the notch filtering during audio demodulation occurs in a situation without any Doppler shift.

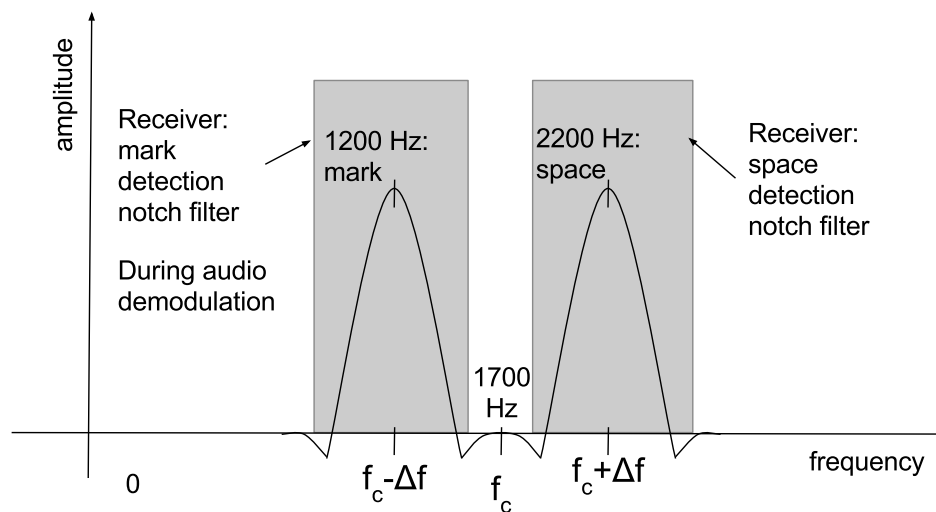
In Fig. 2.35b there are cases where the demodulated RF signal would relay to the audio demodulator (software or hardware defined) a Doppler shifted tone where much of the signal power is contained within frequencies outside the mark or space notch filters. In these images



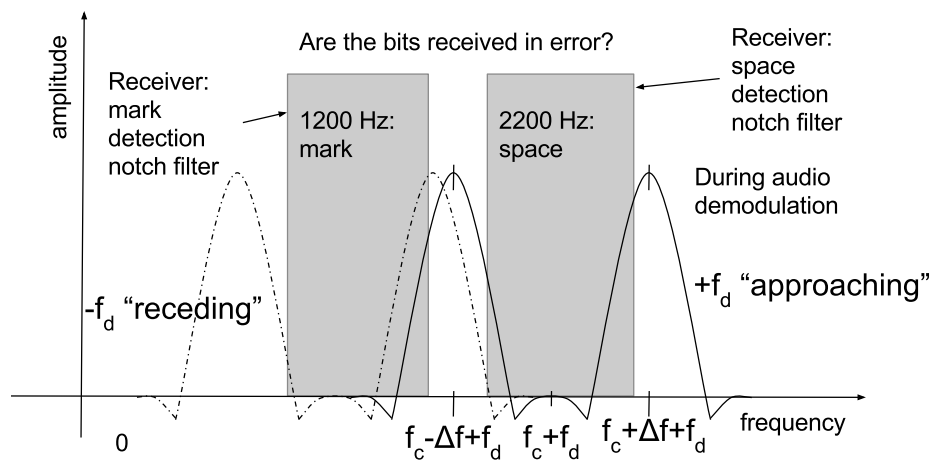
**Fig. 2.34:** Doppler shift impact vs APRS channel notch filter.

$\Delta f$  represents the baseband centre frequency deviation to create the mark (1) and space (0) digital symbols from the audio tones.

It would be impractical to simulate this problem space entirely within a software environment. This situation is further compounded by the possible variations of hardware and software used by the independently operated I-Gates. To do this one could either build a synthetic environment model or build a physical local area network. Rather than rely on detailed model development and selection of an appropriate environment, we can use existing Commercial Off the Shelf (COTS) software and equipment to create a small network and determine if these Doppler induced errors represent an insurmountable problem.



(a) No Doppler induced bit errors.



(b) Doppler induced bit errors.

**Fig. 2.35:** Potential Doppler shift could induce errors into the bit stream.

## 2.8 Summary

Throughout this chapter we have examined the various features required of a robust and generalized model of a CubeSat communications system, which we have summarized as ExampleSat. To build this model we examined the existing population of CubeSats to see if any trends were apparent. From this population emerged a limited set of both orbital elements and communications systems design options. In addition we have seen how influential the details of a spacecraft's particular orbit can be on its communications capabilities.

To create a virtualized network we explored what the available existing network structures are and the design concepts behind them. Focusing our efforts on the existing APRS network we investigated its topology, the AFSK modulation used, the hardware used, and the AX.25 protocol's packet format as well as the q constructs it uses for its transport layer. Also within this chapter we have looked at some of the metrics of performance that could be used to evaluate the network and its design. This included an exploration of stationarity in signal processing To determine if statistical methods could be used on the aggregated downlink signals.

As part of the thesis we will see how each of these components are expanded upon or directly investigated to create different sets of models that are combined together into a larger discrete event constructive simulation. The outputs of that simulation are analyzed to determine the viability of the proposed virtualized ground station network. In Chapter 3 we will see how the community surveys, custom data extraction software, antenna models, orbital mechanics, and our new ExampleSat-165's details are combined into a synthetic environment for study.

## 3 System Design & Development

In Chapter 3 we will create a simulation of the virtualized ground station network with which ExampleSat-165 can connect. Within this chapter we will discuss the systems and methods used within the research. A number of different simulation tools are evaluated to determine if the system in question could be modeled and thus theoretically possible.

### 3.1 Model & Simulation Design

Some definitions will clarify the remaining sections of this chapter of the thesis. In particular, the differences between a model and a simulation. The NATO Modelling and Simulation Standards Profile [Huiskamp2012] defines a model as:

“A representation of a system, entity, phenomenon, or process. Software models of specific entities are comprised of algorithms and data. A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon or process.”

Similarly the standard goes on to define a simulation as:

“The execution over time of models representing the attributes of one or more entities or processes. Human-in-the-Loop simulations, also known as simulators, are a special class of simulations. A method of implementing a model over time.”

The distinction between these two concepts is significant for this work. We represent entities with models and execute those models across time as simulations. This provides a framework where we can combine different models based on research data to create an overarching simulation. For our virtualized network we have many entities or processes that may need to be modelled separately and then transferred into different simulation tools. It is also important to remember a principle related to the fidelity of models by British mathematician and professor of statistics at the University of Wisconsin, George E.P. Box (October 18, 1919 – March 28, 2013).

“Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful.”

Based on these definitions and principles of fidelity we can narrow down our approach to the design of our virtualized network. One could simply do deterministic calculations to find the link budget for a single satellite to an arbitrary I-Gate used as a ground station. Such a model can be accomplished using spreadsheets as demonstrated by Jan A. King in the AMSAT - IARU Basic Analog Transponder Link\_Budget\_Rev 1.6 and the Annotated Link Model

System version 2.5.3 spreadsheets [King2008, King2010]. These exemplary efforts make plain the intricacies of communications systems link design which may include a simplified form of the orbital mechanics at play. These are complicated and useful models, and one could argue that they are models nested within models.

King's spreadsheets provide the user with the capability to not only model the link budget, but the antenna impedance, spacecraft mass and power budgets. The link budget model also includes calculations about frequencies used, channel bandwidth, transponder gain intermodulation ratio, antenna pointing losses, sky temperature and line losses to mention a few. King's model is designed for a single ground station and assumes it has a Yagi-Uda, Helix, Spiral, Parabolic Reflector, Horn or Phased Array antenna. The spreadsheet has limited capability to interpret calculations related to the satellite's orbit. The link budget calculations are based on the predetermined slant range derived from the ground station's elevation angle to the satellite.

Furthermore, the antenna pointing losses are based on a fixed value estimated by the user. The spreadsheet allows for spacecraft systems design, such as the minimum output power required, but does not permit a look at system performance based on the system design. For these reasons, specialized software tools are required to allow for the combination of both orbital movement of the satellite and telecommunications analysis within a simulation.

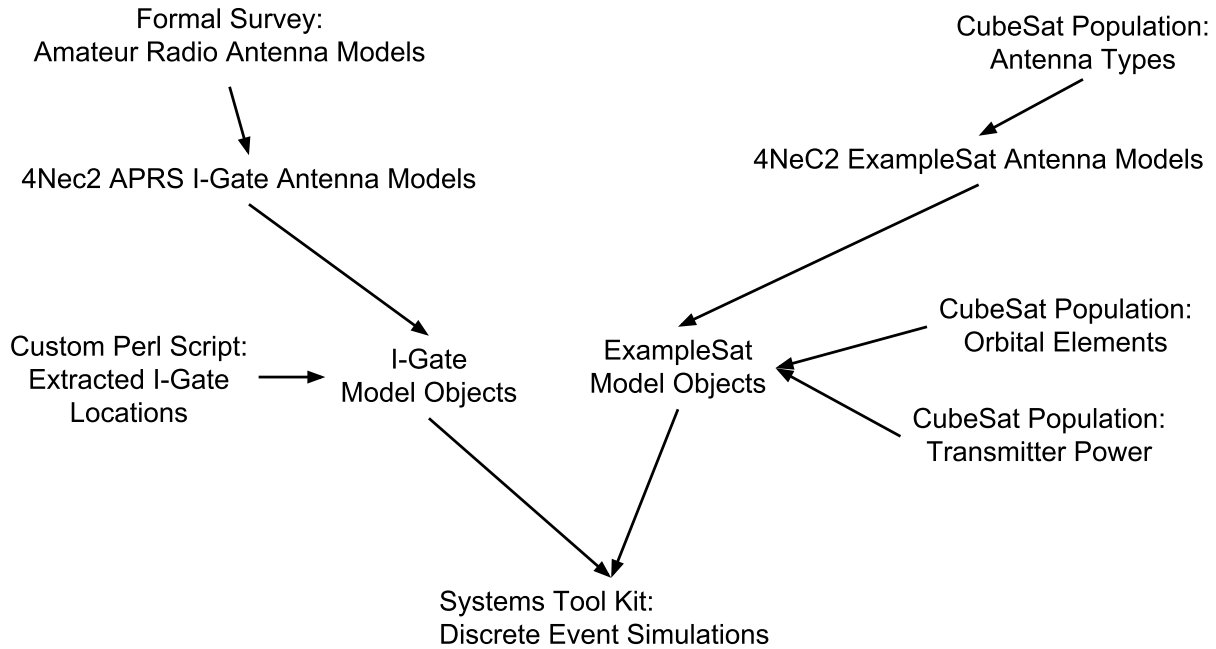
### 3.1.1 Model Composability

To create a useful simulation a number of models must be combined together along with primary research data. We see in Fig. 3.1 how the models and survey data link together:

1. Custom Perl software scripts to extract the I-Gate receiver locations from the APRS Tier 2 server network.
2. A survey of the amateur radio community to determine which antenna designs best represent those used for I-Gates within the APRS network.
3. 4nec2 far field antenna models, selected based on the survey, representing the antennas used by I-Gate within the APRS network.
4. A synthetic environment communications link model using the Systems Tool Kit (STK) software to account for I-Gate receiver locations, antenna far field reception patterns, ExampleSat-165's transmitter details and orbital elements.

To formulate any simulations we must decide what level of fidelity should be represented. We must establish if ExampleSat-165 is capable of communicating with an APRS I-Gate station under some conditions. ExampleSat has a limited transmitter output power of one (1) Watt to apply to an antenna system as established in Fig. 2.24. The available orbits could vary in altitude as derived from the histogram shown in Fig. 2.21.

Based on this data we could choose one or any of the set from 165, 480, 605, and 715 km as median values for multi-modal distribution of common CubeSat orbital altitudes. Progressing forward we will arbitrarily choose 165 km creating ExampleSat-165. Originally this altitude was chosen since it would also induce the most significant rate of change in Doppler shift, described in Sec. 2.7, as ExampleSat-165 passes directly overhead of any single ground station. The implications of Doppler shift upon a receiving I-Gate that does not adjust for it remain as open questions.



**Fig. 3.1:** Logical diagram of the model and simulation set.

### 3.1.2 Amateur Radio Antenna Survey

Because the APRS network is designed, developed, deployed and employed largely by a collection of amateurs who use their own hardware the network is heterogeneous in nature. Some of the equipment is hand made, and some of it is purchased from commercial manufacturers selling products for this enthusiast market. This network heterogeneity also includes the types of antennas used and thus the far field radiation patterns that need to be applied for any models representing the APRS I-Gate network.

We could choose to use our best judgment and arbitrarily select the antenna models. But, to maintain a level of accuracy and generality a survey was conducted of the Amateur Radio community to determine which far field antenna models would be appropriate for use within the simulations. The survey data was only used specifically to narrow down the set of APRS I-Gate ground station antenna models used in an attempt to limit bias of the model selection process. Due to the limited number of respondents to the survey any data from it should not be extrapolated as a representative sample of the community. The data should be considered as an exploratory survey to determine what additional questions could/should be asked of the community.

The nature of the survey is exemplified in Table 3.1. The respondents were first asked a series of questions about their use of I-Gates, the associated antennas and to estimate their station's total received gain. Next they were asked similar questions about their use of Amateur Radio Satellite style stations and the associated antennas. The third section of the survey showed the respondents a series of photos of certain style of antenna and asked them a set of questions about their knowledge or use of that design style of antenna.

Some of the survey respondents estimated the total reception gain, in dB, of their APRS I-Gate ground stations as 6,43,-95,-1,2, and 0. It is likely the response for -95 dB entry was

**Table 3.1:** Survey of I-Gate use.

Do you operate an APRS I-Gate	Count	%
I use it continuously (24/7)	6	37.5
I use it several times a week	1	6.3
I use it a few times a week	3	18.8
I use it a few times a year. (special events, etc.)	0	0
I use it once, possibly twice a year or less	0	0
Its been more than a year since I've used it	1	6.3
Someone else has borrowed it from me	0	0
I do not recall the last time I used it	0	0
I've never actually used it	5	31.3

erroneous.

**Table 3.2:** Survey of AMSAT/OSCAR ground station use.

Do you operate a satellite ground station?	Count	%
Yes	7	43.8
No	9	56.3

**Table 3.3:** Survey of AMSAT/OSCAR ground station antennas.

If you do operate a satellite ground station, can you describe its antenna?	Count	%
J-Pole type	0	0
Whip type	1	14.3
Yagi-Uda type	6	45.7
4 element ground plane with a vertical whip type	0	0
Other	2	28.6

Additionally, some of the survey respondents estimated the total reception gain of their AMSAT/OSCAR ground stations, in dBi, as: 6, 7, 10, 14.39, and 24.

From a space systems development standpoint it is interesting that there is a difference in use between the APRS I-Gates use and the AMSAT/OSCAR station use. Since satellite operations are typically 24 hour, 365 days per year process systems designers desire a ground station that is capable of communicating continuously while the spacecraft is within LOS. APRS I-Gate use for those that did operate one was 81.8% continuously whereas of the those who used an AMSAT capable station only one operated it continuously. 42.9% responded that they occasionally use their AMSAT ground station.

It would be improper to extrapolate to the general population of amateur radio operators due to the limited sample size, but one could conclude that APRS and I-Gate stations provide



utility for those contributing to the network. In contrast its not clear what utility or benefit a volunteer receives from downlinking CubeSat data to be passed along to a university of other institution. Due to the nature of this pilot study some responses are unexpected. For example seven people responded that they have a satellite ground station, while eight supplied responses to “If you operate a satellite ground station, how often is it in use?” because not all of the questions were compulsory. Additional details for all of the survey questions are available in Appendix D.

The third section of the formal survey showed the respondents a picture of an antenna’s wire diagram as well as the emitted far field with a small text description of the antenna and its typical use. They were then asked a series of questions with various rating scales. A selection of the responses for the same antenna design shown in Fig. 2.15 are outlined below:

Do you own an antenna like the one shown above?

Response	Count	Percent
Yes	12	66.7
No	6	33.3

If you own an antenna like the one show, how often do you use it?

Response	Count	Percent
I use it continuously (24/7)	3	27.3
I use it several times a week	2	18.2
I use it a few times a week	0	0
I use it a few times a year. (special events, etc.)	3	27.3
I use it once, possibly twice a year or less	0	0
Its been more than a year since I’ve used it	3	27.3
Someone else has borrowed it from me	0	0
I’ve never actually used it	0	0
I do not recall the last time I used it	0	0

Have you seen the above antenna used for an APRS I-Gate?

Response	Count	Percent
Yes, I’ve seen it used for an APRS I-Gate	7	58.3
No, I have not seen it used for an APRS I-Gate	5	41.7

Was an antenna like the one shown above used for APRS (at your station or someone else’s)?

Response	Count	Percent
Yes	7	58.3
No	4	33.3
I’m not sure	1	8.3

How common is the antenna design shown above?

Response	Count	Percent
Very Common	8	53.3
Common	7	46.7
Not Sure	0	0
Rare	0	0
Very Rare	0	0

Based on the results of the community survey Table 3.4 shows the models that were chosen to represent different ground station antennas. The Yagi-Uda antenna is used for comparative purposes as a conventional, purpose built AMSAT/OSCAR compatible ground station.

**Table 3.4:** Antennas models selected with input from the community survey.

Antenna Design	4nec2 Model Name	% Population	Count in Network Model
Ground Plane	2mGP	63.3	534
Whip	GP144-435	18.2	154
J-Pole	J-Ant 144	18.2	154
Yagi-Uda	216CP_SatComMod	NA	NA

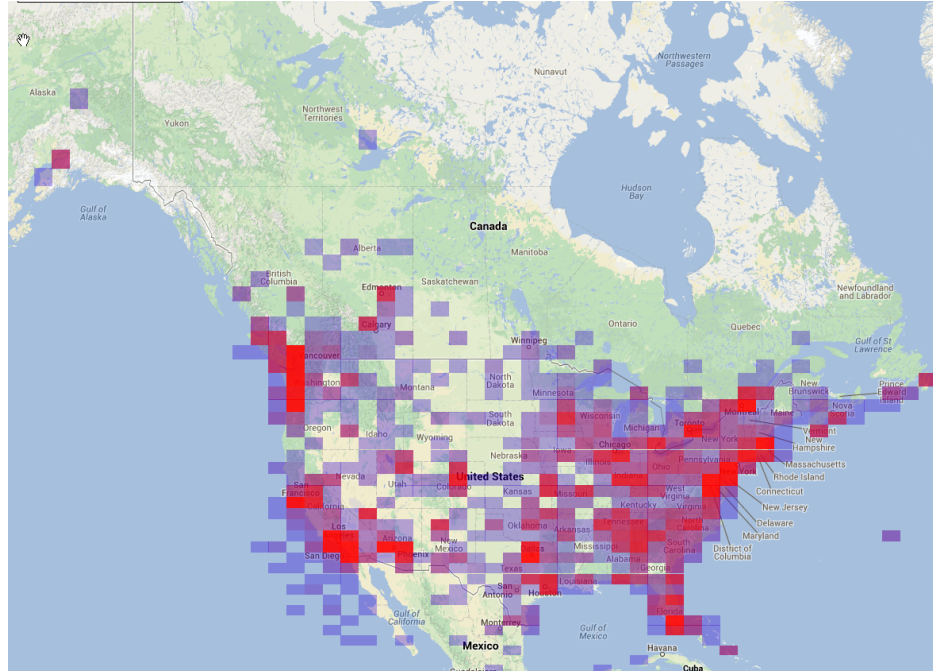
The distribution of responses for the antenna models used for I-Gates demonstrate that the distribution for particular antenna model is not equal. The total set of I-Gate antenna models is loaded to mimic the overall communities preference for the ground plane design. All of this survey data was used in combination with the data collected directly from real world transmitted APRS packets as shown in Sec. 3.1.3.

### 3.1.3 I-Gate Locations

For a simulation of the virtualized ground station to be useful it needs to represent not only the telecommunications systems in question, and the orbital mechanics of the satellite, but the locations of the I-Gates themselves. At first glance this data would seem to be easily obtained from various websites such as APRS.fi as shown in Fig. 3.2 and Fig. 2.8.

Although the locations are visible in both figures, they do not provide the location data itself in an exportable and reusable format. A set of custom Perl scripts was created to extract all of the unique I-Gate call signs and GPS locations across North America. Details of these scripts follows in Sec. 4.2, while the source code is located in Sec. C.1. Based on data provided from the APRS Tier 2 network all of the locations and call signs could then be imported into STK as shown in Fig. 3.3 for further link budget analysis.

This model of I-Gate nodes represents a realistic set of widely distributed nodes participating in an RF network on a what can be considered a continuous basis. Based on statistics previously available from APRS.fi's database (in May 2016 the data visualization graphics were unavailable) it is estimated that 380000 unique APRS nodes are contributing packets to the Tier 2 servers every 30 minutes. So far, no previous research has been published related to characterizing this publicly viewable network system with global reach.



**Fig. 3.2:** APRS node “heat map” of North America. From: APRS.fi

## 3.2 Modelling Tool Selection & Implementation

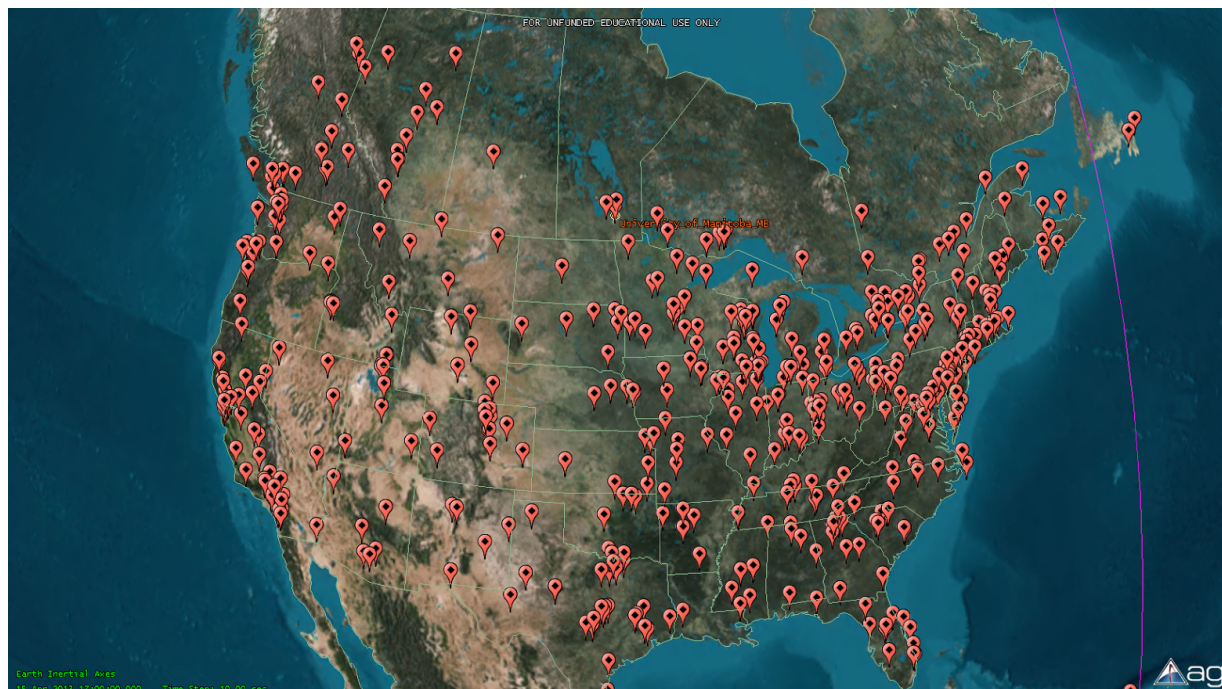
There are a number of antenna modelling software programs available to determine the far field radiation patterns for both the I-Gate ground station and the ExampleSat-165’s antennas. To appreciate the complexity of the software choices available for computational electromagnetics we will look at two techniques of numerical analysis for antenna modelling are prevalent within the field. The Method of Moments and the Finite Element Method.

The numerical technique of using weighted residuals is referred to as the Method of Moments (MoM). Much of the software using the phrase MoM is actually using the Boundary Element Method (BEM) since it applies the moment method to the solution of the integral equations for the surface of the antenna material. Effectively, objects are replaced by free space and the calculated surface currents. This technique is used in particular for modelling Perfect Electric Conductor (PEC) designs such as classic “thin wire” antennas[Hubing2008].

The Finite Element Method (FEM) divides the volume of the antenna into equal shapes which are then meshed together so that the surface currents can be calculated. This allows for differing materials to be analyzed within the same overall model. As such this technique is used for inhomogeneous configurations such as printed circuit boards, but does not succeed with PEC “thin wire” antennas compared to MoM.

### 3.2.1 HFSS

High Frequency Structure Simulator (HFSS) is commercial modelling software from Ansys Inc. based on a FEM solver focused on antennas and RF electronic circuit elements. This software bases its Partial Differential Equations on the connections between tetrahedron



**Fig. 3.3:** APRS I-Gate node map for much of North America.

shaped volumes to solve for the near and far field radiation patterns. The HFSS software package offers FEM, Time Domain FEM, MoM (BEM), Physical Optics, and some hybrid methods. In personal communications with researchers, this author has found that HFSS is sometimes preferred within the computational electromagnetics community for its accuracy. Although this software package is very capable it is excessive for the requirements of this thesis.

### 3.2.2 FEKO

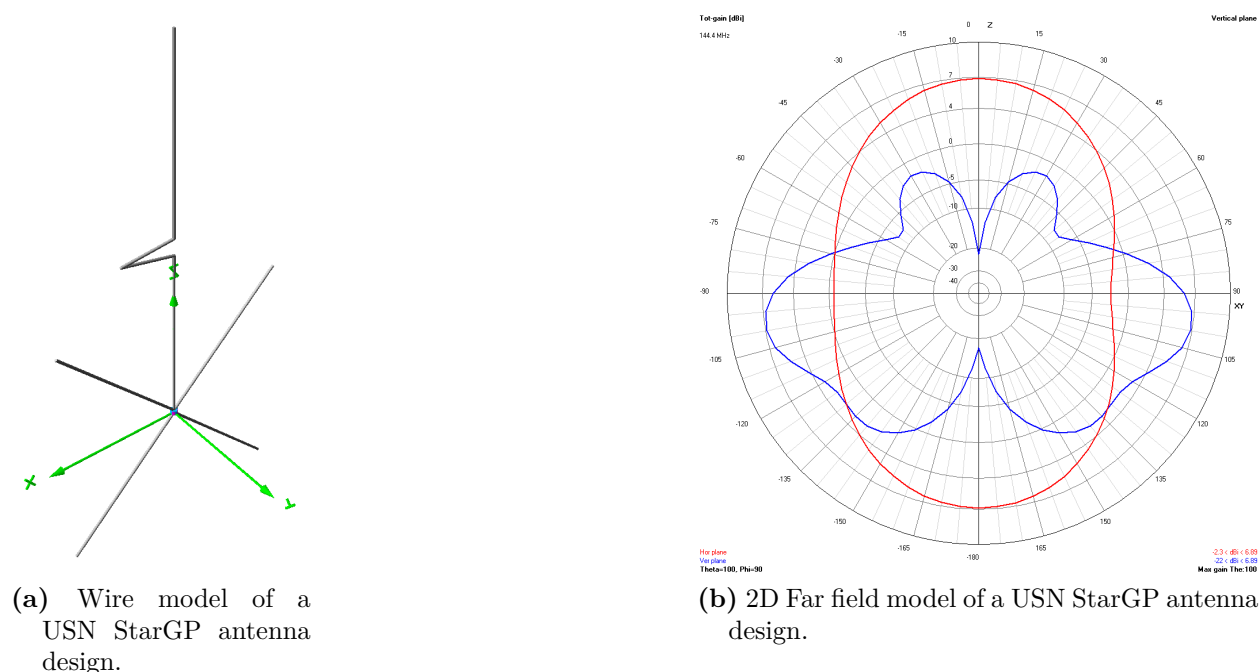
The FEKO software package offered by Altair provides a number of numerical methods including hybrid techniques. These methods include MoM (BEM), FEM, Finite Difference Time Domain, Physical Optics, Geometrical Optics, Uniform Theory of Diffraction and Multi-level Fast Multipole Methods. Although this software package is very capable it is excessive for the requirements of this thesis.

### 3.2.3 4nec2

4nec2 [Voors2016] is a free Microsoft Windows based program using BEM for its numerical analysis. Because it is both free and uses only BEM it is well suited to antenna modelling within the Amateur Radio community due of the lower frequencies used by much of the amateur radio community. The antenna systems typically used by this community involve copper wire, aluminum tubes or aluminum rods for elements. This naturally leads to the use of BEM which is well suited to dipoles, and Yagi-Uda antennas demonstrated as popular by the community survey. The software itself is hosted by [www.qsl.net](http://www.qsl.net) which is a website

dedicated to connecting amateur radio operators around the world. The software has proven popular because its design is purpose built for this community and the type of antennas they use. The software includes a library of antenna models that have been traditionally used within the community, as well as some that are more experimental in nature.

By altering the antenna designs that come included with 4nec2 much of the development time required was reduced. Additionally the more complex numerical methods are not required for the PEC antennas used within the selected amateur radio bands. Some of the types of outputs available from 4nec2 are displayed in Fig. 3.4. In the first sub-figure a 3D diagram showing the wires within the antenna design. The second Sub-figure shows a 2D plot of the horizontal and vertical far field radiation patterns. The latter plots are what many in the amateur radio would be familiar with viewing as commercially available antennas often include similar plots within their advertising specifications.

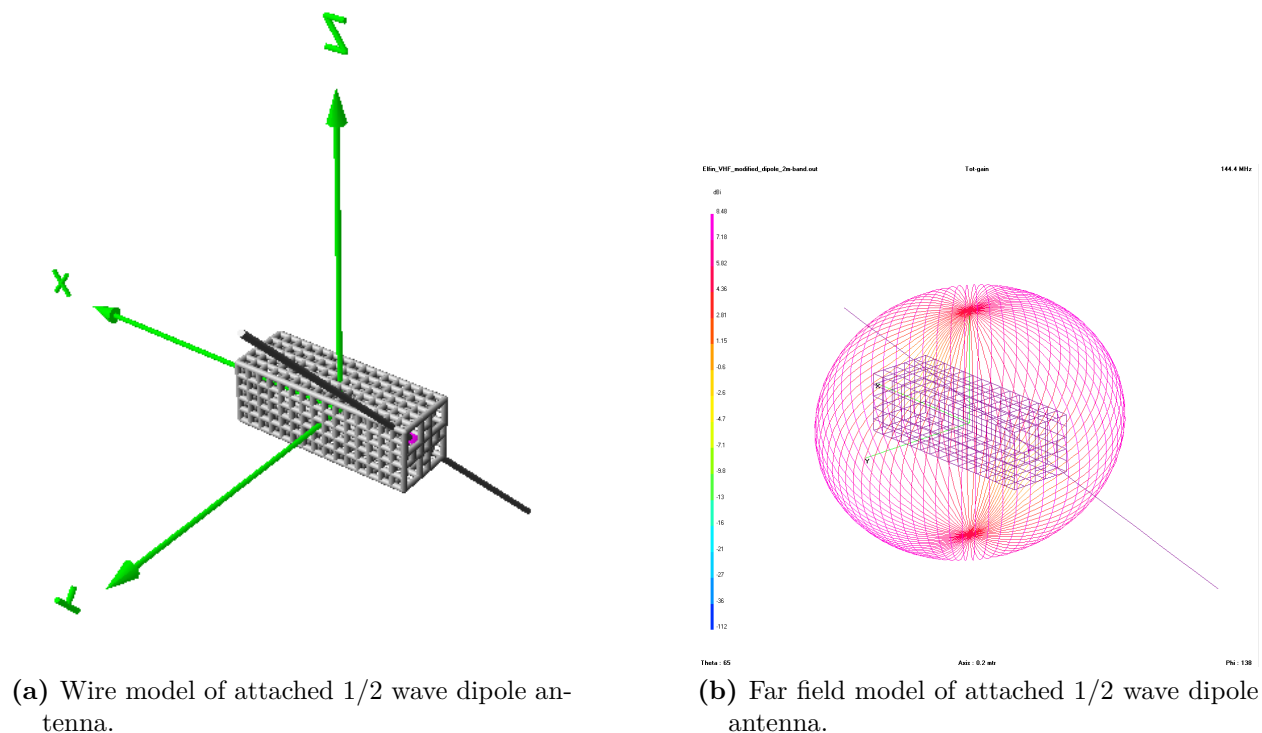


**Fig. 3.4:** Example ground station antenna models in 4nec2.

4nec2 was also used to create the far field antenna patterns for ExampleSat-165 and its variants. As we can see in Fig. 3.5 the first Sub-figure shows the wire model of the antenna as well as ExampleSat-165's structure. Since 4nec2 is designed with BEM numerical analysis in mind, the structure is represented by a series of connected wires to form a 10 cm by 10 cm by 30 cm structure. The model in Fig. 3.5a shows a representative physical structure for a  $1/2$  wave dipole attached to the frame. Assuming that the wavelength of interest is approximately two meters, the near field interactions with the frame can be considered a metal sheet. This gives rise to the far field diagram shown in Fig. 3.5b which accounts for both the attached dipole and the interaction with the satellite's metal structure. The models shown in Fig. 2.28 were based on modifications from those used by the development team of the (ELFIN) CubeSat project with the University of California, Los Angeles Institute of



Geophysics and Planetary Physics [IGPP2014].



**Fig. 3.5:** ExampleSat-165 antenna models rendered in 4nec2.

Designers will need to adjust the antenna elements within 4nec2 to align the boresight with the Z-Axis and then output the desired Phi Theta dB far field pattern table. This is because of the nature of axis definition within STK when compared to 4nec2. STK uses the z-axis as a reference frame pointing towards the central body within the simulation. In contrast, 4nec2 uses the z-axis to refer to an arbitrary “up” direction away from the ground plane. This extra step is only required for the directional antenna beams and not the set of omnidirectional antennas used as their boresights are already designed to point upwards from the Earth’s surface or an associated ground plane.

4nec2 can be used to generate an exportable plot of points using the Phi/Theta pattern style of output. However, by not installing the recommended GNUplot software an error message is generated, then the output is dumped to a text file. This allowed for the creation of a set of comma separated value text files for each antenna that included the total number of points within the set and the far field radiated power in dB for each angular combination of phi at theta degrees. Combined, this set of steps enabled the export of a series of PEC far field antenna models into Analytical Graphics Inc.’s Systems Tool Kit simulation software.

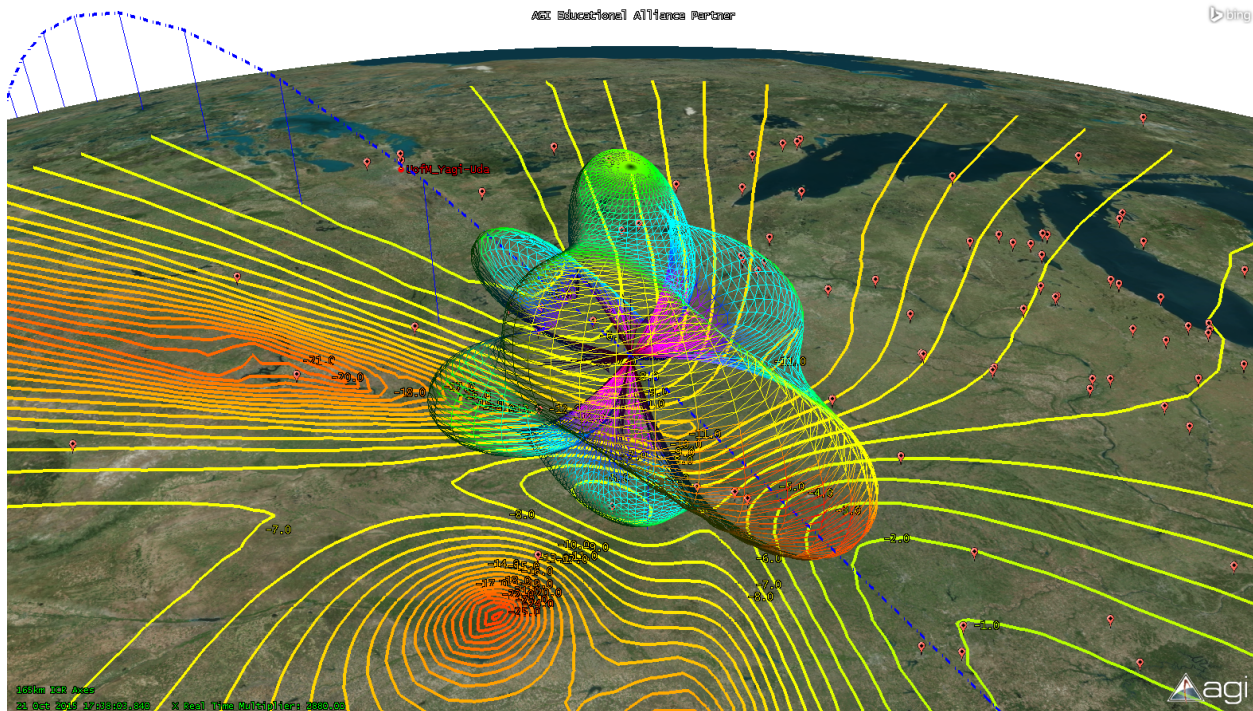
### 3.2.4 Systems Tool Kit

To analyze the link budget, specifically the  $E_b/N_0$ , from ExampleSat-165 as received by 842 different I-Gate locations over a period of 24 hours, the conventional approach of using spreadsheets for deterministic analysis would be impractical. A simulation tool to combine

the set of antenna models and the orbital mechanics across time is required. Two robust software systems are available for this purpose, Analytical Graphics Inc.'s Systems Tool Kit (STK) [AGI2016] and a.i.-solution's FreeFlyer [a.i.solutions2016]. STK was chosen largely based on this author's familiarity with the simulation tool and knowledge of its telecommunications analysis capabilities.

STK is based on a geometry engine to calculate the time-dynamic position of objects along with the relationships between them under a user defined set of constraints. This windows based software was originally developed for the aerospace community for use in orbit analysis and access calculations. Over time the application areas for the software has expanded to include calculations related to topics such as radar, electronic warfare, conjunction analysis, space situational awareness, sensor coverage, unmanned air systems, real time data visualizations, spacecraft maneuvering and extra-planetary missions, aircraft and missile mission planning, and GPS dilution of precision for example.

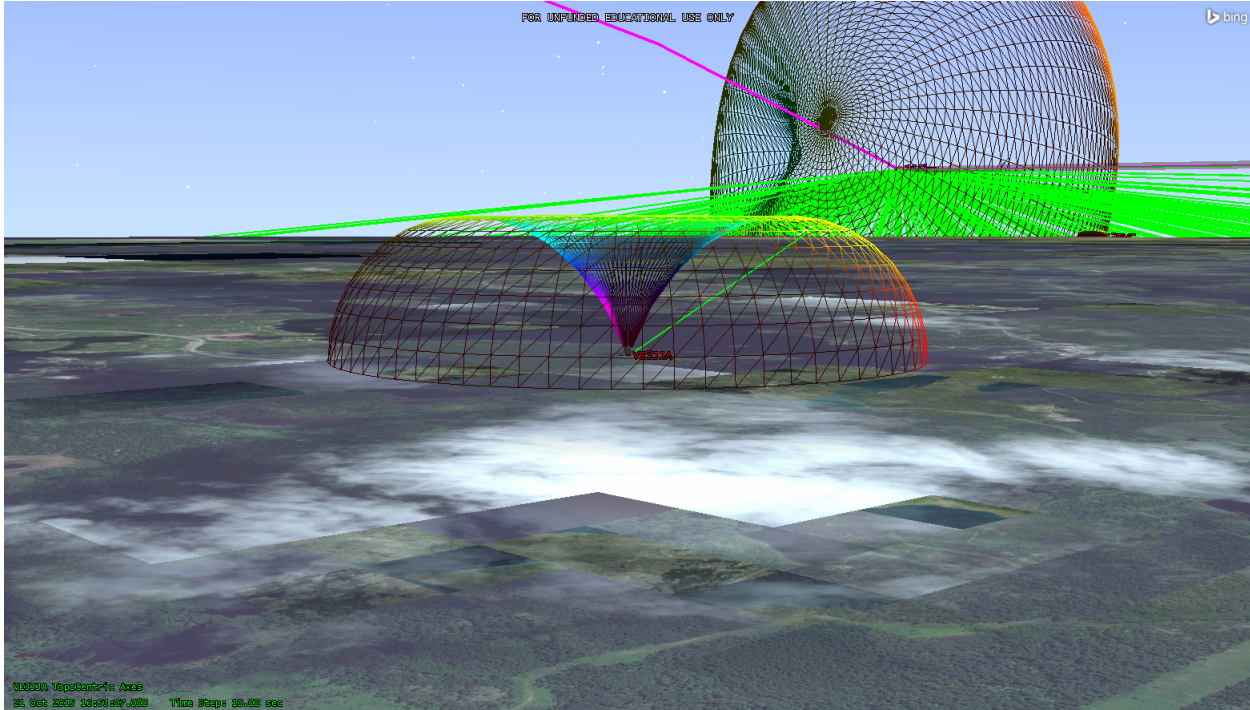
The ability to combine both geophysical locations with antenna patterns and orbital positions is exemplified by STK in the following images. The image shown in Fig. 3.6 shows one possible variant of ExampleSat-165, the far field radiation pattern of a turnstile antenna with corresponding footprint projected on the Earth's surface as contour lines. The corresponding lines relate to the received signal strength in dB.



**Fig. 3.6:** ExampleSat-165 with a “turnstile” far field antenna pattern in orbit above North American APRS I-Gate locations.

Displayed in Fig. 3.7 is the far field antenna pattern for both an I-Gate, using a ground plane antenna, and for ExampleSat-165 using a dipole antenna. The green lines represent a line-of-sight link between the available ground station and ExampleSat-165. The magenta arc to the upper middle of the image running through ExampleSat-165 represents the current

trajectory of the spacecraft.



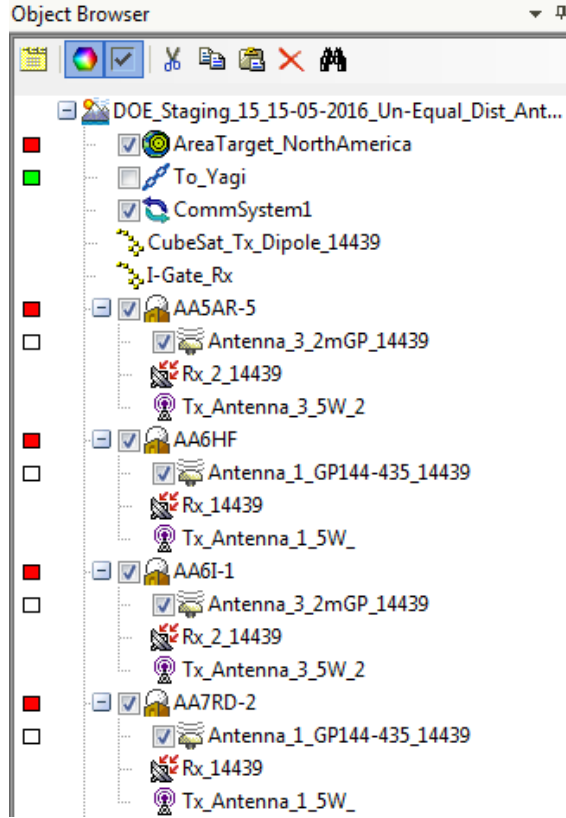
**Fig. 3.7:** Far field patterns of ExampleSat-165's and a single I-Gate.

STK simulations are based on the principle of scenarios for development and planning purposes. In this way, the simulation tool is used to create a series of different scenarios for comparative analysis. Scenarios are designed around a designated timeline for the analysis as well as the epoch time described in Sec.2.5.1.1. Each scenario contains a number of different modeled objects such as the Earth as a central body. As we see in Fig. 3.8a and Fig. 3.8b each object can have a series of child objects. These can include antenna patterns, transmitter details, receiver details and antenna pointing characteristics.

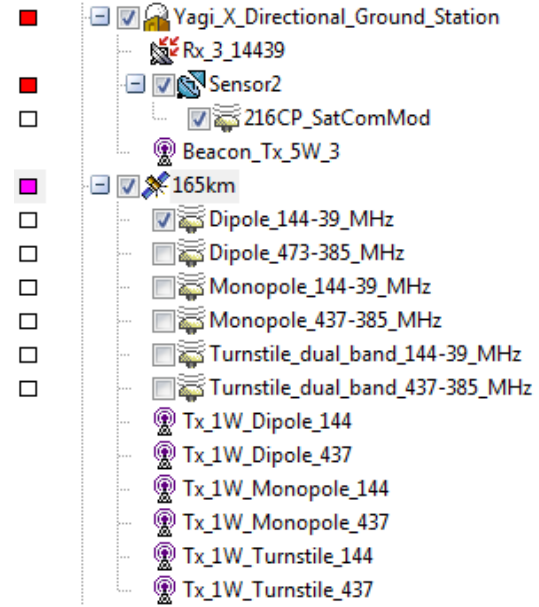
Ground stations are represented on the surface of the Department of Defence World Geodetic System (WGS84) [NIMA2000] which includes a standardized coordinate system, a reference ellipsoid (called the datum) and the gravitational equipotential surface. By using the same reference ellipsoid as used by the GPS constellation we avoid treating the Earth as an idealized sphere. The fact that the Earth is an oblate spheroid, due to its axial rotation, creates a difference of 21 km (0.335 %) between the equatorial axis and the polar axis. This could have a significant impact on the viability of any link budget model derived from a simple sphere. While the orbit may be nearly circular, as described in Sec.2.5.2, the Earth and thus the surface that the ground stations are resting on are not. Let us look at an example to illustrate this. Given a transmitter of zero gain (0 dB), a receiver of zero gain (0 dB), a frequency of 144.39 MHz and a distance of 21 km we can determine the free space path loss associated with this delta as if ExampleSat-165 had passed over the North pole and the prime meridian at the equator.

$$L_S [dB] = 20 \log_{10} (d) + 20 \log_{10} (f_c) - 147.55 - G_t - G_r \quad (3.1)$$





(a) I-Gate ground stations with receivers, transmitters and antennas.



(b) Yagi-Uda station and ExampleSat-165 with receivers, transmitters and antennas.

**Fig. 3.8:** Object browser view.

$$L_S [dB] = 20 \log_{10} (21) + 20 \log_{10} (144.39) - 147.55 - 0 - 0$$

$$L_F [dB] = 102.1$$

Where  $L_S$  is the Free Space Path Loss (FSPL) in dB,  $d$  is the distance in km,  $f_c$  is the centre frequency of the transmitter in MHz,  $G_t$  is the gain of the transmitter, and  $G_r$  is the gain of the receiver.

When the margin for a successful communications link could be as low as 4 dB, a delta of 102.1 dB due a modelling assumption is significant. It's understood that the likelihood of such an extreme situation occurring with a CubeSat mission design are limited but an added loss of several dB may be possible. This was another strong motivator for using a professional modelling tool rather than a custom coded solution. Furthermore, simulations created within STK can be shared with other institutions to expedite their mission development.

### 3.3 R programming language

As mentioned in Sec. 2.6.2 the R programming language [Ihaka2016] was used to generate the White and Brown Noise data series via the R Studio [Allaire2016] IDE. Specifically, each series was generated by the TK95 function from the {RobPer} package[?] which uses the power law function where  $1/f^0$  creates White Noise and  $1/f^2$  creates Brown Noise.

The R language was chosen based on several key factors. The first was that its open source and thus freely available (as in “*gratis*” or “free beer”). This includes added packages that contain callable mathematical functions all of which are created by the user community. This user community is robust [smith2012r] with a number of websites dedicated to showing how to use the programming language and its capabilities [CRAN2016, Analytics2016]. R was developed specifically with statistical computing in mind and is actively used by statisticians and data miners[Muenchen2015]. As such R has built in statistical analysis functions such as linear and non-linear regression, ANOVA, clustering, support for structured data, and is actively used for time series analysis. R offers a straightforward way to load a data set and begin exploring it with function calls with no cost. An added feature is the R and the associated R Studio are capable of creating high quality scalable vector graphics for print publications. R and the associated R Studio were used extensively within Chapter 5 as part of the analysis,

### 3.4 I-Gate Emulation

In Sec. 2.7.1 the problem of Doppler induced bit errors was introduced. To determine if the Doppler shift would induce errors via an I-Gate’s audio demodulation notch filter as shown in Fig. 2.35 and experiment was designed based on a software defined radio. GNU Radio is an open source software package for experimentation with software defined radio [GNURadio2016]. Its graphical interface is familiar to those who’ve used MATLAB’s Simulink software in that signal processing blocks are arranged and linked together to accomplish some computational task. In a similar way GNU Radio uses this style of graphical interface to enable the user to link together both audio domain and RF domain signal processing tasks. The output of one block can be input to another. We can see the graphical user interface and an initial design experiment with APRS in Fig. 3.9.

This software package can connect via USB to communicate the hardware such as the dongle shown in Fig. 2.6.

The layout in Fig. 3.10 shows the design a local area network APRS system to test the practicality of the overall thesis concept. In particular the scenario was created to see if the GNU Radio software could be configured to transmit APRS packets.

The premise behind the experiment was to leverage the capability of a commercially available software defined radio with the open source GNU Radio software. With the help of Matt Woelk, for his software development expertise, the mathematical models to describe the impact of Doppler shift, as described in Sec. 2.7 (in particular Eq. 2.17 and Eq. 2.20), were crafted into a custom Python script to be used as a signal processing block within GNU Radio. This would in effect become a satellite channel emulation tool where the calculated Doppler shift could be emulated by imputing the orbital elements for ExampleSat-165’s while

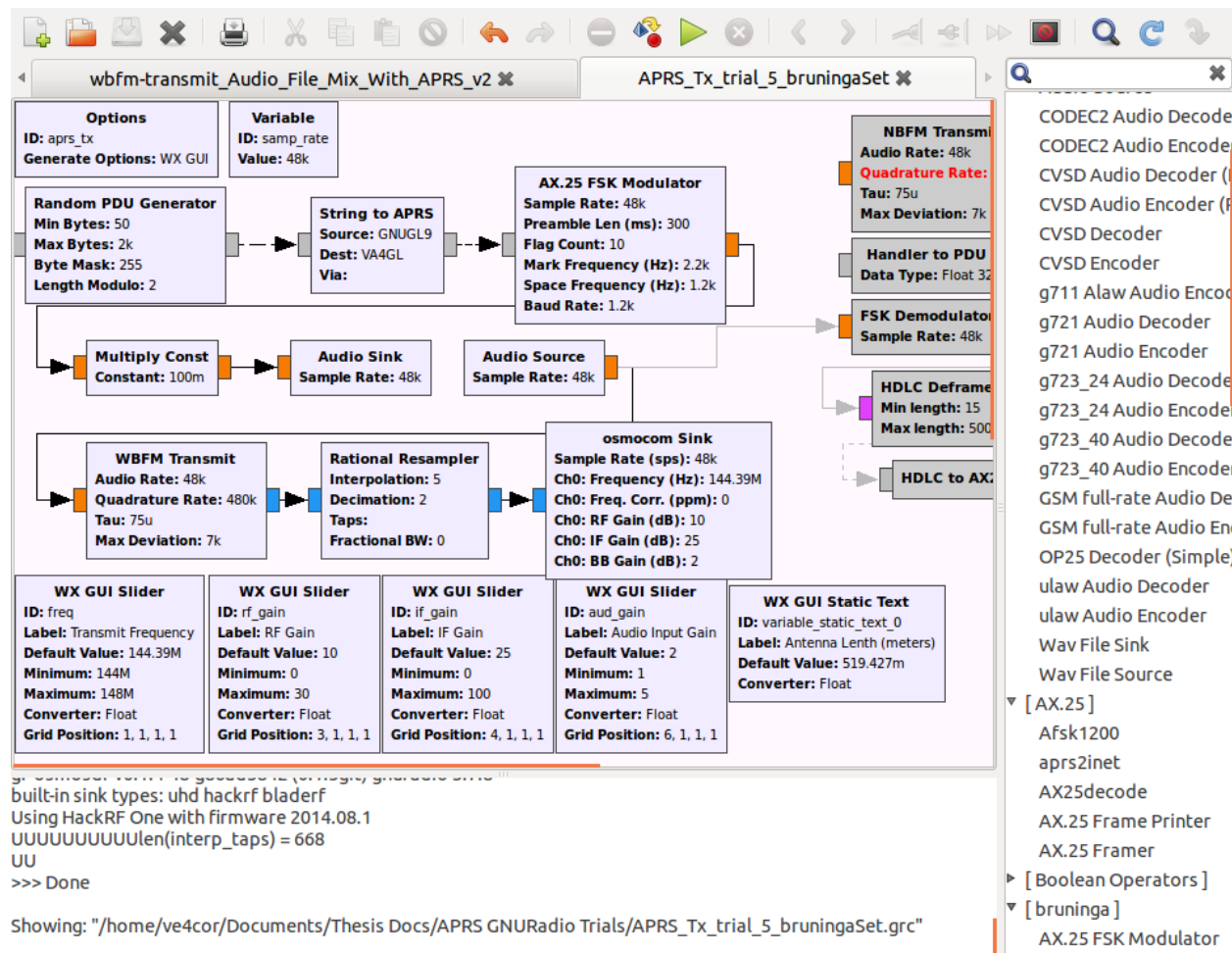


Fig. 3.9: GNU Radio graphical interface.

maintaining the generality of geophysical positioning.

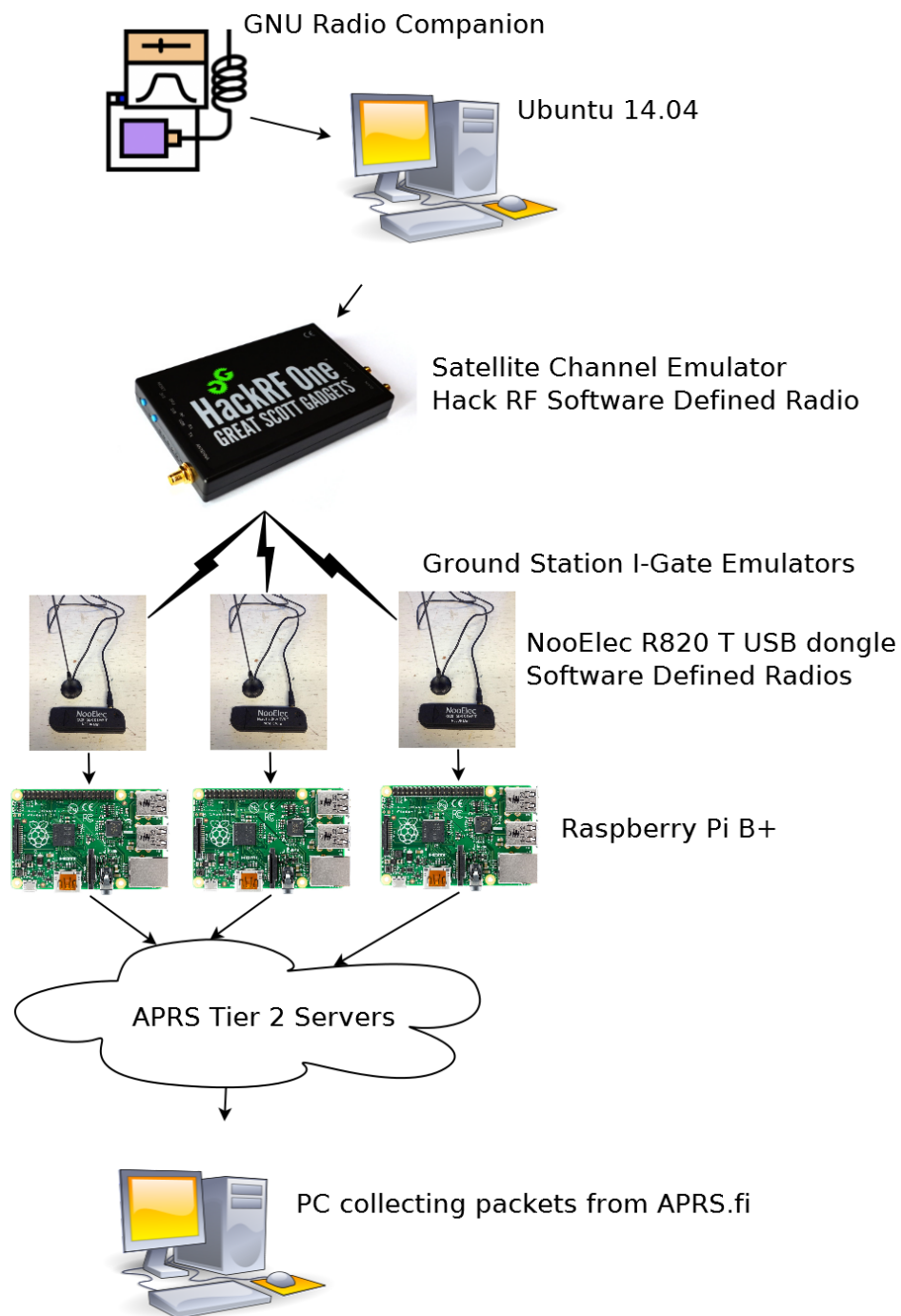
With this combination of hardware and software the emulator would allow the experimenter to alter the set of orbital elements for ExampleSat-165 as well as the downlink frequency or modulation technique simply by altering values within the signal processing blocks. This design would allow for repeated trials where the rate of change in slant range could be predictably controlled based on the precise orbital mechanics of the satellite's downlink. Attenuation due to atmospheric or ionospheric conditions could also be applied by adding additional signal processing block. Overall the design allows for a realistic transmission of APRS packets to be received by realistic hardware as used by the amateur radio community.

For this experiment the NooElec NESDR Mini SDR & DVB-T USB dongle was paired with a Raspberry Pi B+ [Foundation2016] to act as the ground station. The dongle acted as the RF receiver and the Raspberry Pi was loaded with software to act as the TNC as well as the overall I-Gate. ExampleSat-165's rapidly moving transmitter was to be emulated by Doppler adjusting the transmission frequency of a HackRF from Great Scott Gadgets [HackRFOne] during the period of a single orbital pass.

The Mini SDR dongle is capable of receiving frequencies from 25 MHz up to 1750 MHz. The HackRF One is a half-duplex transceiver operating from 1 MHz to 6 GHz with 8 bit quadrature samples for both I and Q.

Unfortunately this experimental setup was not viable in time for this thesis. The hardware as purchased and tested but the customized software module would not function within GNU Radio. In part this is because of the experimental and developmental nature of the GNU Radio software package. At the time its documentation was lacking making the creation of “out of tree modules”, modules not included with the official distribution, very difficult.

As such this aspect of the research was pursued but ultimately shelved after three months of development due to time constraints. It’s still felt that this method of satellite channel emulation will yield significant results since the total hardware cost is under \$500 CDN not including the PC and the platform is flexible enough to realistically emulate both ends of the communications link.



**Fig. 3.10:** Local network physical diagram with SDR Satellite Channel Emulator and APRS I-Gates.

## 3.5 Summary

Throughout Chapter 3 we have seen how a set of models were created using primary research data and different software tools to enable us to create a single simulation that can run these models across time. This process involved collecting survey data and transmitted packet data from the amateur radio community its self. This data was used to ensure the relevance and or practicality of the models used and not as measures of prediction for the community itself. No statistical inferences were determined about the population at large from the collected survey data. The packet data and I-Gate location information represents real world values for the APRS network over that period of time. This is in effect the most complete model of the APRS I-Gate network so far without assumed positions or over simplifications. We may also speculate that the overall simulation and the models it's comprised of represent the first attempt at conducting network analysis of not only the APRS network but of any RF network of this large scale.

With regards to far field antenna modeling software, there are a number of different products available each with different strengths and weaknesses largely dependent on the core computational methods used. The choice of using 4nec2 was based on both the prebuilt model set within the frequency bands of interest but also because of its relative simplicity compared to both HFSS and FEKO. Additionally the capabilities within the latter two are not required within the discrete event simulation tool as well see in Chapter 4.

We have also explored an experimental setup for satellite channel emulation. Unfortunately the Doppler shift out of tree module needed to complete this experiment was not operable in time for this publication.

## 4 System Implementation

In Chapter 3 we discussed the overarching systems design approach of combining different models together into a single discrete event simulation tool. In this chapter we will explore the specifics about how this simulation was built and the types of output data that was collected from it. While Chapter 3 discussed in general terms the modelling tools to be used and the objects to be created within the simulation, this chapter will dive into the specifics of what's needed for each model object and the discrete event simulation within STK.

### 4.1 Scenario Overview

To study the interaction of satellite based APRS transmissions received by the I-Gate stations within the terrestrial APRS network a simulation representing the spacecraft on-orbit relative to the ground stations is required. Such a simulation allows us to study the interactions between the spacecraft and the APRS network over a large area such as a portion of the North American continent.

A key issue within any CubeSat communications downlink is the duration of the LOS access time as described in Sec.2.1. This duration limits the total amount of information that the satellite operator can retrieve from their science payloads. To establish the boundary conditions for this problems space we will design our simulations around a “near worst case scenario”. This choice provides a lower boundary with which future designs can be improved upon.

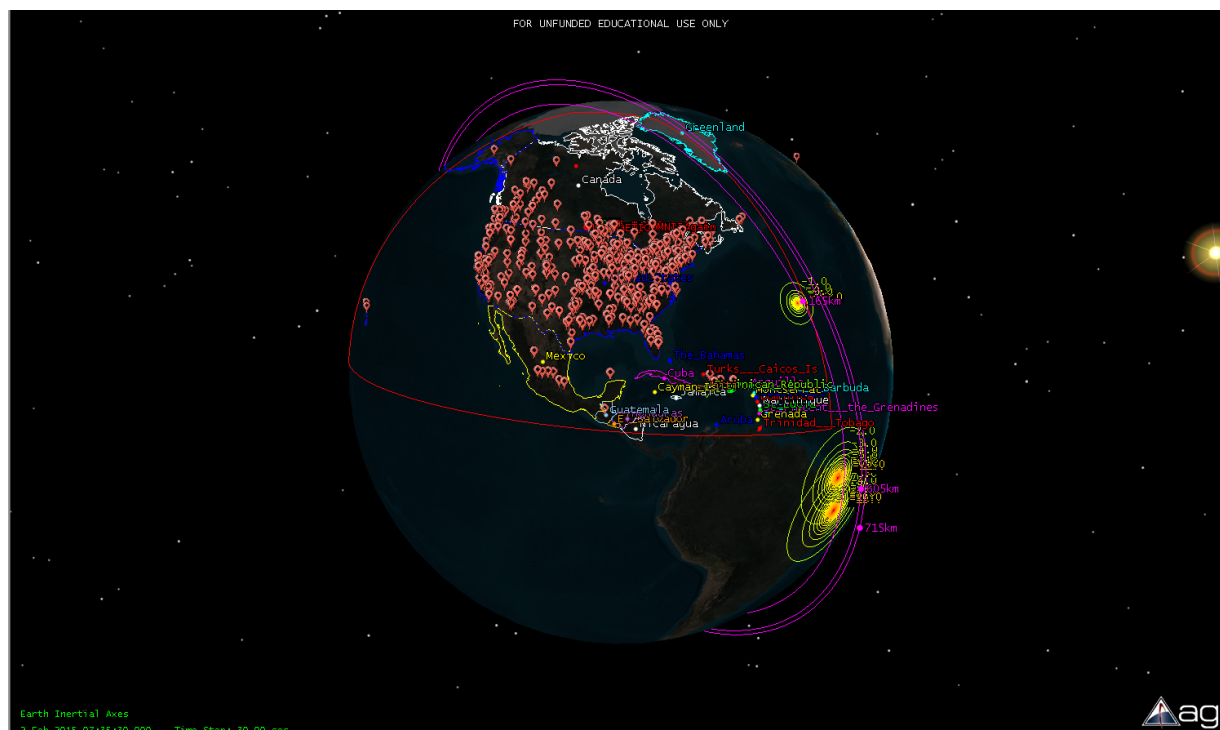
This “near worst case scenario” would occur if ExampleSat is:

- at its lowest median orbital altitude (165 Km)
- at its lowest transmission baud rate (1200 baud)
- at its lowest mean transmission power (1 W)

We should also note that all of the data collected from the scenario is over a 24 hour period starting from the set epoch date October 21 2015 05:00:00.000 Coordinated Universal Time Greenwich Meridian (UTC).

Overall each of the parameters for our test case CubeSat were described throughout Chapter 2. This creates a single scenario file that can asses a set of different experimental schemes if desired. For this stage of the research we will use two core schema under test. Within Fig. 4.1 we can see how a number of different test schema are possible.

Several different satellite orbits are shown as purple rings, each one indicating a different altitude. Below each satellite are contour lines representing the communications footprint from the spacecraft to the ground. The I-Gate stations can be seen dotted across North America,



**Fig. 4.1:** STK Scenario with multiple ExampleSats, and I-Gate locations.

which is surrounded by a red boundary line. The first schema is to have ExampleSat-165 downlinking to VE3UM which is configured with a conventional steerable Yagi-Uda antenna pointed at the spacecraft during LOS access times. The second is to have ExampleSat-165 downlinking to the combined set of APRS I-Gate stations across North America when each individual ground station has LOS access time. This situation is exemplified in Fig. 4.2 in which we can see a large toroidal shape for ExampleSat-165's far field antenna pattern. The green lines emanating from the centre represent LOS access times from the single spacecraft to the many visible I-Gates.

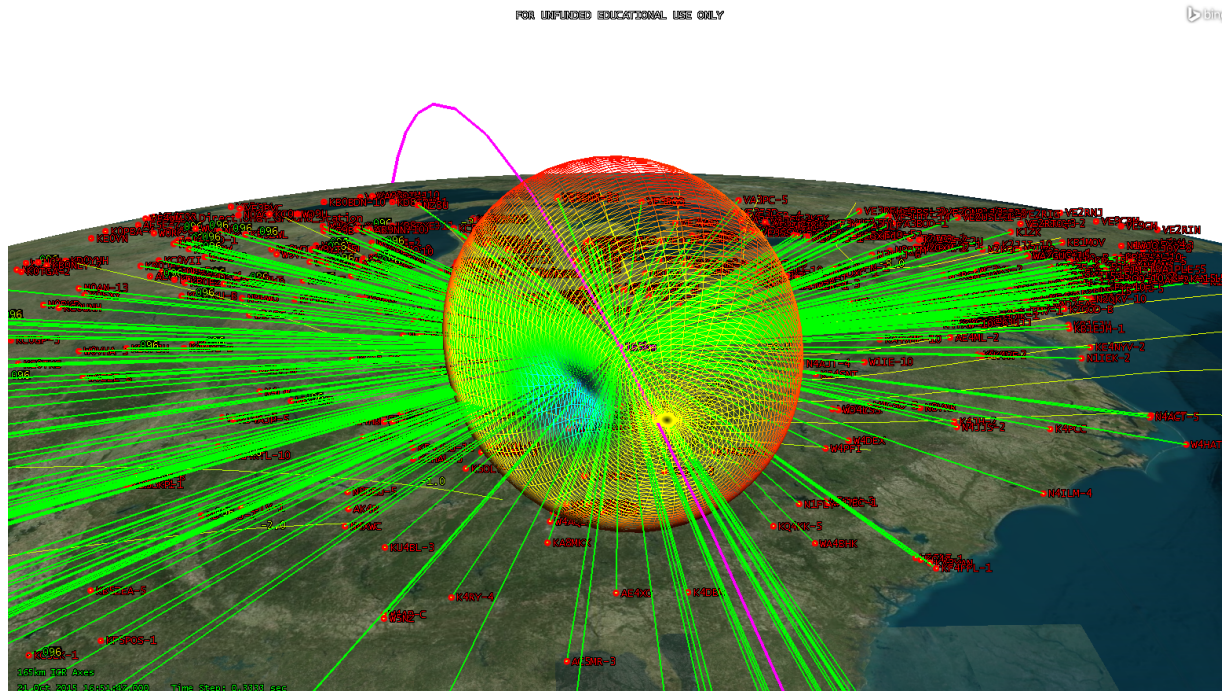
## 4.2 I-Gate Location extraction

To create a more accurate representation of a virtualized ground station network using the APRS I-Gates we need to know the position of those I-Gates across North America. Fortunately the APRS network provides globally unique call sign identifiers for every node on the network and each node typically provides its GPS coordinates as part of its message content.

To collect the I-Gate call signs and locations three custom Perl scripts were created with the assistance of Rob Barwell whom is a friend and colleague with a software development background.. The source code for these scripts are available in Sec. C.1.

The code was based on the Finnish APRS Parser (Fabulous APRS Parser) Perl module originally written by Tapio Sokura for the APRS.fi website[Sokur2012]. Raw APRS data packets are collected from the Tier 2 Network through an application program interface





**Fig. 4.2:** APRS I-Gate locations across North America imported via KML into STK simulation software.

(API) provided by APRS.fi [APRS.fi2016]. The packets are parsed by several conditions. Packets must follow the q-construct code format as described in Sec. 2.4.7. If they contain either “qAR” or “qAr” any text between them and the character “:” within the packet structure must be the call sign of the I-Gate that relayed the packet from the RF domain onto the internet.

Next the software script compares the call sign for those that correspond with a list of prefix’s for all the possible licensed amateur radio call sign variants within North America. This method has the advantage that call signs are administered by each country and that globally every call sign must be unique to particular operator. Additionally amateur radio operators are encouraged to use distinct suffixes, based on their own call signs, if they use multiple APRS stations. This assists in ensuring uniqueness within the data set of collected I-Gate call signs.

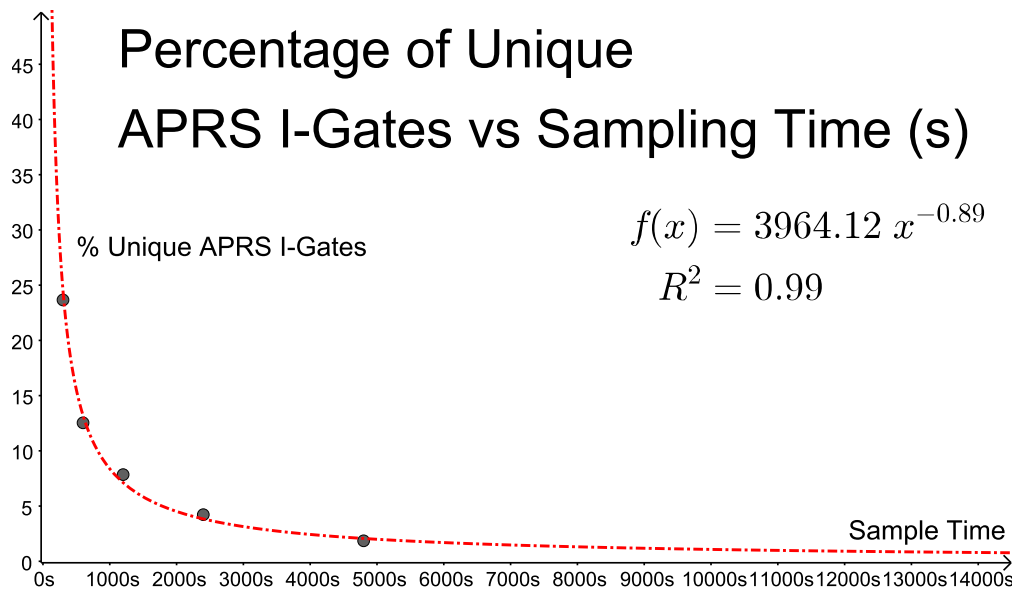
The set of call signs are then queried through APRS.fi to determine each stations latitude and longitude location. This set is then geographically filtered to ensure that the stations are actually within the North American region. Combined this set of scripts creates a list of unique call signs with latitude and longitude for stations that are only I-Gates and are likely to be at fixed locations rather than visitors to North America. Note however that from an overall systems integration viewpoint, it does not matter if the I-Gate is fixed or mobile, native to the region or visiting, as long as its capable of relaying packets from the RF domain to the Tier 2 servers.

The collection of I-Gate call signs, and therefore locations, is derived from the publicly available data set that is assembled as the I-Gates themselves relay data from the RF domain

to the Tier 2 network servers. To ensure that the number of call signs within the data set is representative of the population the raw data feed is collected over time. Since each I-Gate is operated independently they can each have differing intervals in which they report packets to the Tier 2 servers. While sampling the raw packets, as time progresses, less unique call signs meeting the North American I-Gate criteria appear. As we see in Table 4.1 and in Fig. 4.3 the time required to sample the raw data stream follows a predictable trend.

**Table 4.1:** Sample table of Uniqueness for APRS I-Gates.

Time (s)	Total Call Signs	Unique Call Signs	% Unique
300	2333	552	23.66
600	5201	652	12.53
1200	9150	718	7.85
2400	17839	753	4.22
4800	46552	863	1.85



**Fig. 4.3:** Plot of Percent Uniqueness of I-Gates vs Time.

By following the trend shown in Fig. 4.3 the sample time was adjusted to collect a predicted 98% of the population of unique APRS I-Gate call signs and locations. With 863 unique call signs this set of I-Gates was filtered down to 842 which could be mapped accurately within other simulation tools as shown in Fig. 3.3.

## 4.3 STK Simulations

### 4.3.1 STK Scenario Setup

The overall simulation requires global definitions to work within. The analysis period of 24 hours was chosen arbitrarily to start at 21 October 2015 at 05:00:00:000 UTCG. The epoch time, as described in Sec. 2.5.1, was also chosen to start at this point. The discrete calculation increment or “Step Size” for animations and visualizations was chosen arbitrarily at ten (10) seconds. A smaller Step Size is analogous to increasing the frequency of sampling from the simulation and incurring resulting the computational costs. Individual analysis components such as link budgets can be selected as having different step sizes within the discrete event simulation.

It was initially attempted to utilize the basic Digital Terrain Elevation Data (DTED) to increase the fidelity of the simulation as terrain masking could then limit the ability for numbers of I-Gate ground stations to have LOS with ExampleSat-165 during its orbital passes. With the large number of ground stations, each with individual link budget calculations this proved to be too computationally intensive for the available computer hardware at the time. The simulation would stall the operating system or the computer would crash. STK now supports distributed computing capabilities which could alleviate this issues. However, we can question if this would have a statistically significant impact on the overall analysis as it may not limit the number of I-Gate ground stations to below a threshold needed to archive a combined stationary  $E_b/N_0$ .

The overall RF environment within STK has a number of selectable capabilities. This can include an active interference environment which is why transmitters other than the one for ExampleSat-165 were included in the set of STK objects described in Sec. 4.3.2. An attempt was made to collect the mean number of received APRS packets to all of the I-Gates represented in the simulation. This would have required creating a co-located interference source for each I-Gate to represent the nearby RF traffic attempting to use the I-Gate at semi-random intervals. Additionally, this proved to be computationally difficult for the computers available at the time. This also indicates the utility of using the APRS network for network analysis. It represents a set of networked nodes that make all of their data traffic publicly available.

For calculations related to RF reception that are dependent on propagation within the atmosphere as well as the ionosphere several settings were left as the typical default values. The Earth Temperature used was 16.85 degrees C as an average temperature. Rain, cloud and fog are represented by the ITU-R P618-10 “Propagation data and prediction methods required for the design of Earth-space telecommunication systems” [ITU2001] as a reference model. Atmospheric attenuation similarly uses the ITU-R-P676-5 “Attenuation by atmospheric gases” [ITU2013].

We should note that according to the ITU:

Attenuation by atmospheric gases which are entirely caused by absorption depends mainly on frequency, elevation angle, altitude above sea level and water vapor density (absolute humidity). At frequencies below 10 GHz, it may normally be neglected [ITU2001].

As a secondary reference we should also point out Paffett *et al.* whom authored “VHF-Band Interference Avoidance for Next-Generation Small Satellites” further support that VHF is not impacted by atmospheric and ionosphere propagation [Paffett1998]. Realistically the atmospheric attenuation below 10 GHz is not zero but is between 5 and 6 orders of logarithmic magnitude lower than the free space path loss. For this model we can assume that the atmospheric loss (in dB) negligible due to the use of the 144 MHz frequency band. Detailed information for the atmospheric attenuation of frequencies above 1 GHz are commonly available at via the ITU. All of the above is to state that while the attenuation due to water (vapor and molecules), oxygen, hydrogen and other gasses is assumed to be limited at the frequencies of interest these scalars are still calculated within the simulation. The caveat being that the commonly used propagation models are limited in their capacity when studying anything under one (1) GHz.

Related to the propagation within the VHF and UHF bands the tropospheric scintillation is also taken into account. In particular because we are interested in the VHF band we included calculations for deep fade which may accrue as depending on the angel of incidence from ExampleSat-165 to an I-Gate do to reflections from the Earth’s surface back into the antenna causing significant destructive interference.

Additionally we can state that STK itself is not designed for large numbers of objects. The graphical user interface does not directly accommodate the selection of 842 ground station receivers within many of its submenus or subwindows. For example to determine the link budget values required manual clicking of each station’s receiver which involved three steps each. This process takes about a half hour to complete manually once proficient which delayed the analysis. Its possible to create scripts to activate tasks within STK, but such user specific software development was not possible in time.

### 4.3.2 STK Objects

Scenarios within STK are coordinated though objects added to the scenario with conditions and constraints applied across time. To get a gauge to the level of fidelity for the overall simulation a brief description of the objects represented within is illustrative. Each object typically has a set of child objects that represent different associated models that are dependent on the parent as shown in Fig. 3.8.

**Area\_Target:** A region was created in the scenario environment to encompass North America. This was to ensure that I-Gates within the simulation were actually within the North American region.

**I-Gates:** Each I-Gate is represented by its location upon the Earth’s surface with a latitude and longitude based on the WGS84 datum.

**Antenna:** Each I-Gate has an attached antenna from one of three different models of which an example is shown in Table 3.4 and represented by a far field antenna pattern as described in Sec. 3.2.3. All of these antennas were specified as having a design frequency of 144.39 MHz and the far field models were generated as such. All of the the I-Gate antennas were positioned as 2.5 m above the surface to represent a median height for its installation. Its important to remember that

these I-Gate antennas are not steerable at targets in any particular direction apart from their fixed far field radiation pattern.

**Receiver:** Each I-Gate has an attached receiver tuned to 144.39 MHz and to not “Auto-Track”. This prevented the simulation from allowing the I-Gate receives from tuning to adjust for Doppler Shift. The linked antenna described above is used with polarized set to vertical. The corresponds to the typical terrestrial antenna orientation. The System Noise Temperature was accounted for including the thermally induced transmission line noise from the antenna to the Low Noise Amplifier (LNA) and the LNA to the receiver transmission line. These temperature values were set at 290 K with the LNA noise figure itself at one (1) dB. Antenna Noise included in the simulation accounts for noise sources such as the solar, atmospheric, rain, tropospheric scintillation and cosmic background radiation. The demodulation is set to auto-select such that the transmitter during a link chooses the demodulation technique. Interestingly this could allow for one to use varied cross-demodulation methods if not set to automatic. No additional gains or losses were added to the receivers.

**Transmitter:** To enable future studies each I-Gate has an attached transmitter. The tuning frequency was also set to 144.39 MHz with a bit rate of 1200 bits/s. Based on data collected via and informal community survey the transmit power was set to 5 W as shown in Table B.7. The antenna settings are the same as the receiver attached to the same parent object. The modulator chosen was FSK with equidistant upper and lower bandwidth limits of +3.5 kHz and -3.5 kHz respectively giving a total bandwidth of 7 kHz. This transmitter allows for testing of interference methods and self-induced jamming.

**ExampleSat-165:** As previously described in Sec. 2.5.1 the satellite was selected as having an eccentricity of approximately zero with both an apogee and perigee of 165 km. This was chosen as a worst case scenario for non-linearity of the received  $E_b/N_0$  as well as the Doppler Rate. The inclination was chosen at a CubeSat population median of 97.7924 deg. The argument of perigee is 0 deg. based on the eccentricity, the RAAN is arbitrarily 204 deg.

**Antenna:** The antenna attached to ExampleSat-165 was also given a design frequency of 144.39 MHz. It was based on an external antenna pattern file provided by far field calculations from 4nec2 as shown in Fig. 2.28.

**Transmitter:** A complex transmitter model is attached to ExampleSat-165. It is tuned to 144.39 MHz with an output power of 0 dBW (1 Watt) at a data rate of 1200 bits/s based on the CubeSat community survey data from Sec. 2.5.3. The attached antenna is described above. Unlike the ground station it does not calculate polarization, this is done by the receiving end of the link. The modulator chosen was FSK with equidistant upper and lower bandwidth limits of +3.5 kHz and -3.5 kHz respectively giving a total bandwidth of 7 kHz. This corresponds to a discrete event simulation version of the AFSK modulation as described in Sec. 2.4.2.

**Receiver:** Since the simulation is only interested in downlinks from space to the ground, receivers were not attached to ExampleSat-165.

**AMSAT/OSCAR\_station:** To get a relative qualitative comparison between a conventionally designed ground station and the virtualized ground station network, a facility was added to the scenario which has similar designs to a typical AMSAT / OSCAR station. This object is visible in Sub-fig. Fig.3.8, note the ground station is name Yagi\_X\_Directional\_Ground\_Station within the simulation to aid in differentiating it from the co-located VE4UM I-Gate station.

**Sensor:** To make the antenna track ExampleSat-165 as it orbits, a sensor object is attached to the AMSAT/OSCAR\_station. This sensor is defined as having a narrow a simple conic half angle of 0.1 deg. Then it is given a pointing definition of targeted at ExampleSat-165. This creates a simple cone emanating from the ground station toward ExampleSat-165 while it is overhead, enabling the antenna that is attached to the sensor to point at the satellite. As a constraint the sensor is limited to a minimum elevation angle of +10 degrees above the horizon ( $\theta_{min}$ ). This is common within the amateur satellite community as it limits interference noise from local and distant terrestrial emitters.

**Antenna:** The antenna attached to the AMSAT/OSCAR\_station uses an external antenna pattern calculated via 4nec2 as shown in Fig.4.7. This antenna represents a 16 element Yagi-Uda design for the 2 m band with roughly 12 degrees of beamwidth.

**Receiver:** A complex receiver model is attached to the directional ground station. It is tuned to 144.39 MHz. The demodulator is selected as automatic based on ExampleSat-165's transmission. The attached antenna is described above. The antenna to LNA line loss, the LNA gain, the LNA to receiver line loss are all assumed to be 0 dB each. This keeps the scenario somewhat generic in that a specific commercial model number of the receiver itself is not required. Since ExampleSat-165 cannot turn its antenna to face the ground station cross polarization between ExampleSat-165 and AMSAT/OSCAR station may occur. For this reason the AMSAT/OSCAR station's receiver antenna is set to use right-hand circular (RHC) polarization. Using circular polarization is a common tactic with amateur satellite communications community. RHC itself is chosen arbitrarily as the attached antenna far field pattern is nearly equal for all  $\phi$  near the boresight. The demodulator was selected as automatic and is determined by the transmitting ExampleSat-165. The system noise temperature settings is the same as for the I-Gate.

**Transmitter:** A similar transmitter as used by the I-Gates above was attached to the AMSAT/OSCAR ground station for future model comparisons.

This summarizes the different types of objects of interest directly within scenario design. Additional objects include groupings of objects within the scenario called "constellations". For example a constellation of just the I-Gate receivers or just the transmitter on ExampleSat-165. Using constellations enables the user to link together sets of objects for further study in "chains" or sets. Chains within STK are linear connections between objects in a structured order. For example, a ground station in North America, could link to a spacecraft over the Atlantic ocean and subsequently link to another ground station in Europe. This step by

step list of objects forms a chain. From here the software can determine when this chain of objects has the proper LOS access to enable an end to end link.

### 4.3.3 Modulation within STK

The STK simulation environment can handle telecommunications modulation in a number of different ways. When specifying a transmitter system within the software a user can choose from either a predetermined list of modulation types or a customized modulation file. It is possible to define customized modulation types within STK that include the ability to modify commonly available modulation schemes with specific encoding and spectral efficiencies. One can also input a customized Power Spectral Density file. This file requires us to calculate a number of variables so that STK can generate Power Spectral Density (PSD) based on these inputs. As an alternative one can also input the PSD directly.

Based on what was discussed in Sec. 2.4.2 we will consider AFSK as a specialized format of Binary FSK. With real APRS radio equipment this would not be advised as in practice the signal needs to maintain a continuous phase to be interpretable by the receiver. Sudden shifts in phase often characterized by higher order modern FSK modulation techniques, may cause the types of Terminal Node Controller equipment utilized by the community to receive bits in error. However, because STK is a discrete event simulation tool the actual carrier from ExampleSat-165 is not maintained between calculation steps and therefore we can utilize prebuilt FSK modulation types with the simulation and configure one to match our AFSK needs. The details of the modulation as selected within the software is described in Sec. 4.3.2.

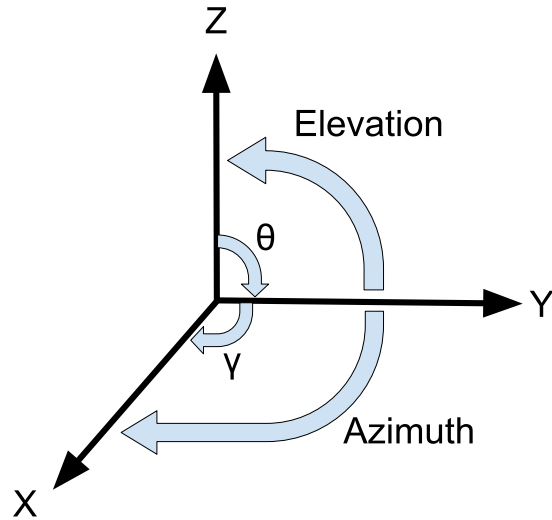
### 4.3.4 STK CommSystem

Within the STK software is an object called a “CommSystem”. This object within the synthetic environment allows the user to include interference sources within the active simulation. By treating nearby transmitters as sources of interference we could simulate how non-satellites transmitting on the same channel can impact the ability for the I-Gates to successfully receive APRS packets from a CubeSat. An initial attempt was made to see the effectiveness of this calculation tool. Its worth recalling that I-Gates themselves also beacon out data across the RF network. In effect the I-Gates become a form of self interference due to their beacon interval. Enabling this feature with 842 ground stations acting as self-interference sources proved to be beyond the computational capacity of the computers available to the researcher at the time. The simulations “crashed” several PCs and laptops forcing the operator to manually restart the simulations. STK now provides functionality for distributing computing, which may elevate this problem in future research.

STK treats  $E_b/N_0$  (dB) as the signal-to-noise ratio at the receiver before demodulation gains or losses. Within the software system the energy per bit  $E_b$ (dB) divided by the total environmental noise  $N_0 + I_0$ (dB) is treated as the energy per bit to (noise + interference) ratio. In this particular case  $N_0$  is equal to the Boltzmann’s constant plus the system temperature (řK), whereas  $I_0$  is the interference power spectral density. Because of the large RF network being simulated  $I_0$  was not set nor used in calculations. It would be possible to perform interference power spectral density calculations for a single I-Gate or AMSAT/OSCAR style station but this was beyond the scope of the thesis at this stage.

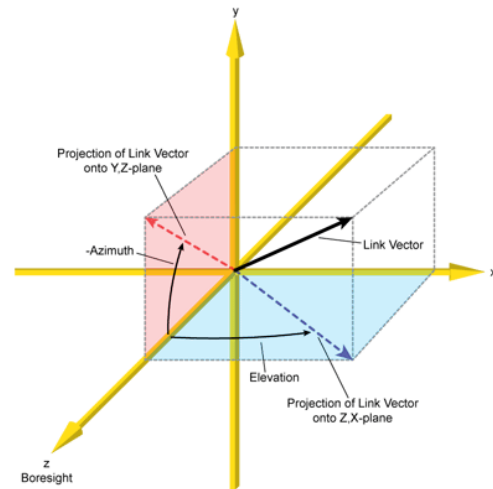
## 4.4 Antenna Models

To create an accurate simulation with more realistic received signal qualities the far field antenna radiation patterns of all the I-Gate receivers, described in Sec. 2.4.5, and ExampleSat-165's transmitters were modelled in 4nec2 and then imported into STK as outlined in Sec. 3.2. Initially this created some problems as the two modelling tools use similar but different co-ordinate systems to orientate the far field radiation patterns.



(a) Typical ThetaPhi coordinate system. After: [Ibanez2012]

AzEl External Antenna Pattern Format



(b) STK's ThetaPhi coordinate system. From: [AGI2016a]

**Fig. 4.4:** Differences in ThetaPhi coordinate systems between modeling and simulation tools.

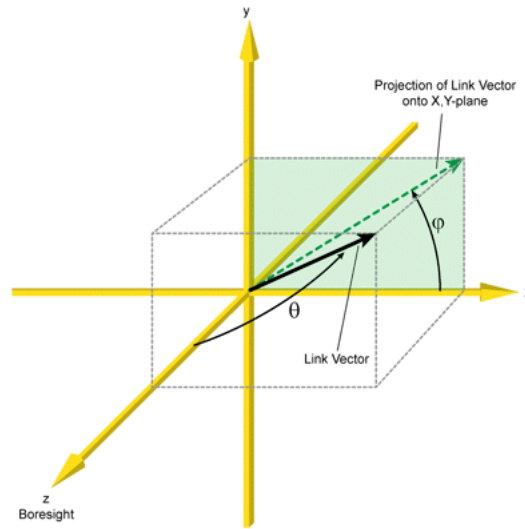
As we can see in Fig. 4.4 that the z-axis for STK's ThetaPhi coordinate system is 90 degrees from what 4nec2 uses. This is because by default STK assumes that the z-axis is along the boresight of the antenna system, whereas 4nec2 considers the z-axis to be perpendicular to the Earth's surface. This interpretable of STK is because from a space based perspective the direction towards the Earth's surface (or centre of mass) changes as the spacecraft moves along its orbit.

This situation is alleviated for specific ground station antenna models by changing the orientation of the far field radiation pattern within STK by 90 degrees. Fortunately this was only required for the Yagi-Uda antenna pattern, as the omnidirectional antennas boresights are symmetrical about the z-axis.

The far field outputs from 4nec2 were in ThetaPhi coordinate system as shown in Fig. 4.5 and rotated manually within STK if required to have the z-axis align between the model types. The 4nec2 output were simple comma separated text files that were modified to be STK compatible. Technically the files were in PhiTheta format listing the Phi angle, followed by the Theta angle and then a gain value in dB for that coordinate point in 3D space. An example can be seen in Table 4.2.



### ThetaPhi External Antenna Pattern Format



**Fig. 4.5:** STK ThetaPhi coordinate system. From: [AGI2016a]

#### 4.4.1 CubeSat Antenna Models

Based on the literature review described in Sec. 2.5.5 three different antenna models were created within 4nec2 for ExampleSat-165's far field antenna patterns.

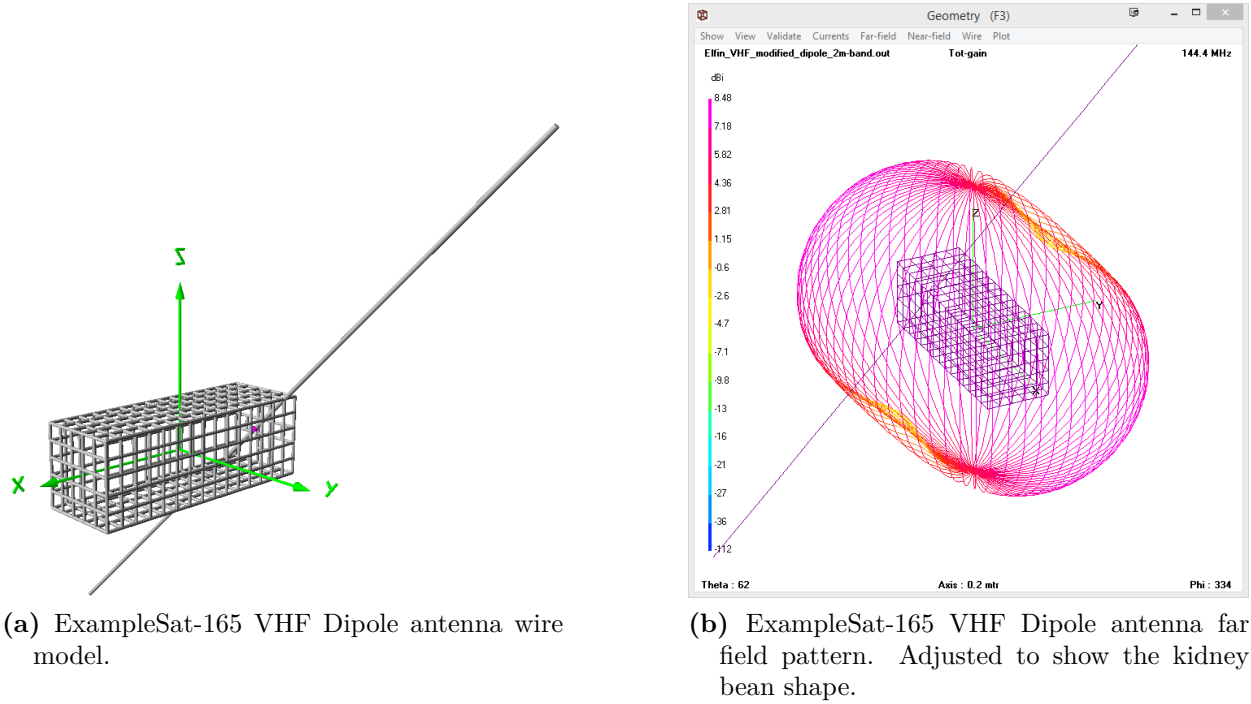
These were evaluated first at 144.39 MHz and then at 437.385 MHz. Single band dipoles and monopoles were modelled as well as dual band turnstile antennas. Although a number of antenna models were created and evaluated at differing frequencies, measures of performance were only collected for the dipole antenna evaluated at 144.39 MHz. This antenna model is illustrated in Fig. 2.28 and has been adjusted to show how the antenna interacts with the CubeSat structure creating the visible “kidney bean” shape of its far field radiation pattern, rather than an idealized toroid. The rest of the antenna models are included for completeness and to enable further research.

**Table 4.2:** Example PhiTheta far field pattern file.

4nec2 Far Field Pattern		
GP144-435 @ APRS Freq.out, Eznec style export on 24-Sep-2014 21:02:59		
Frequency = 144.39 Mhz		
Reference = 0 dBi		
stk.v.10.0		
PhiThetaPattern		
AngleUnits Degrees		
Number of Points 2701		
Pattern Data		
0	-90	-30
...	...	...
10	-90	-30
15	-90	-30
...	...	...
305	45	-1.59

**Table 4.3:** CubeSat antenna models created in 4nec2 and imported into STK.

Antenna Design	Frequency [MHz]
VHF Dipole	144.39
UHF Dipole	437.85
VHF Monopole	144.39
UHF Monopole	437.85
VHF & UHF Turnstile	144.39
VHF & UHF Turnstile	437.85



**Fig. 4.6:** ExampleSat-165's VHF dipole antenna design in 4nec2.

#### 4.4.2 I-Gate Ground Station Antennas

To create a realistic simulation of the large number of APRS I-Gates a series of antenna models were created using 4nec2. As we see in Table 4.4 these models were populated in the scenario based on the amateur radio community survey as described in Sec. 3.1.2.

**Table 4.4:** I-Gate antenna models created in 4nec2 and imported into STK.

Antenna Design	% Popularity	I-Gates	4nec2 Model Name	STK Object Name
Ground Plane	63.3	534	2mGP	Antenna_3
Vertical Whip	18.2	154	GP144-435	Antenna_1
J-Pole	18.2	154	J-Ant144	Antenna_2
Directional Beam	NA	NA	216CP-SatComMod	* Yagi-Uda
Total number of North American I-Gates				846
Total number of I-Gates in simulation				842

\* AMSAT/OSCAR station design used for comparative purposes

##### 4.4.2.1 Yagi-Uda Ground Station Directional Antenna

To simulate a typical AMSAT/OSCAR station a model was created with two sets of orthogonal element arrays. In normal use for communications with a LEO CubeSat such an antenna could be operated in a circularly polarized manner by activating the driven element

in each array at 90 degrees of phase. Doing so allows the operator to partially mitigate the net cross-polarization loss since the CubeSat’s antenna is typically not steerable and has a single polarization orientation relative to the spacecraft’s moment of inertia. Minoli’s textbook “Satellite Engineering in an IPV6 Environment” [Minoli2009] has a good quality and brief introduction to the topic for the neophyte. His text focuses on the commercial satellite communications industry and subsection 2.2.2 on polarization includes a handy guide for what types of polarization are used for each purpose across the industry.

A limitation of the 4nec2 software is that it does not account for polarization of antenna arrays. In Fig. 4.7b the far field radiation pattern is limited to a single active elements orientated along the y-axis. Note that the software still includes the x-axis array in its far field calculations, it simply cannot take a combined set of input phases, and thus orthogonality active arrays, across time to create a median far field calculation. We must make the assumption that since the far field is almost symmetrical about the boresight that this would have limited impact on the model. Additionally, any polarization effects are accounted for within the STK simulation software. However, this is predicated on an accurate representation of the antenna’s far field.

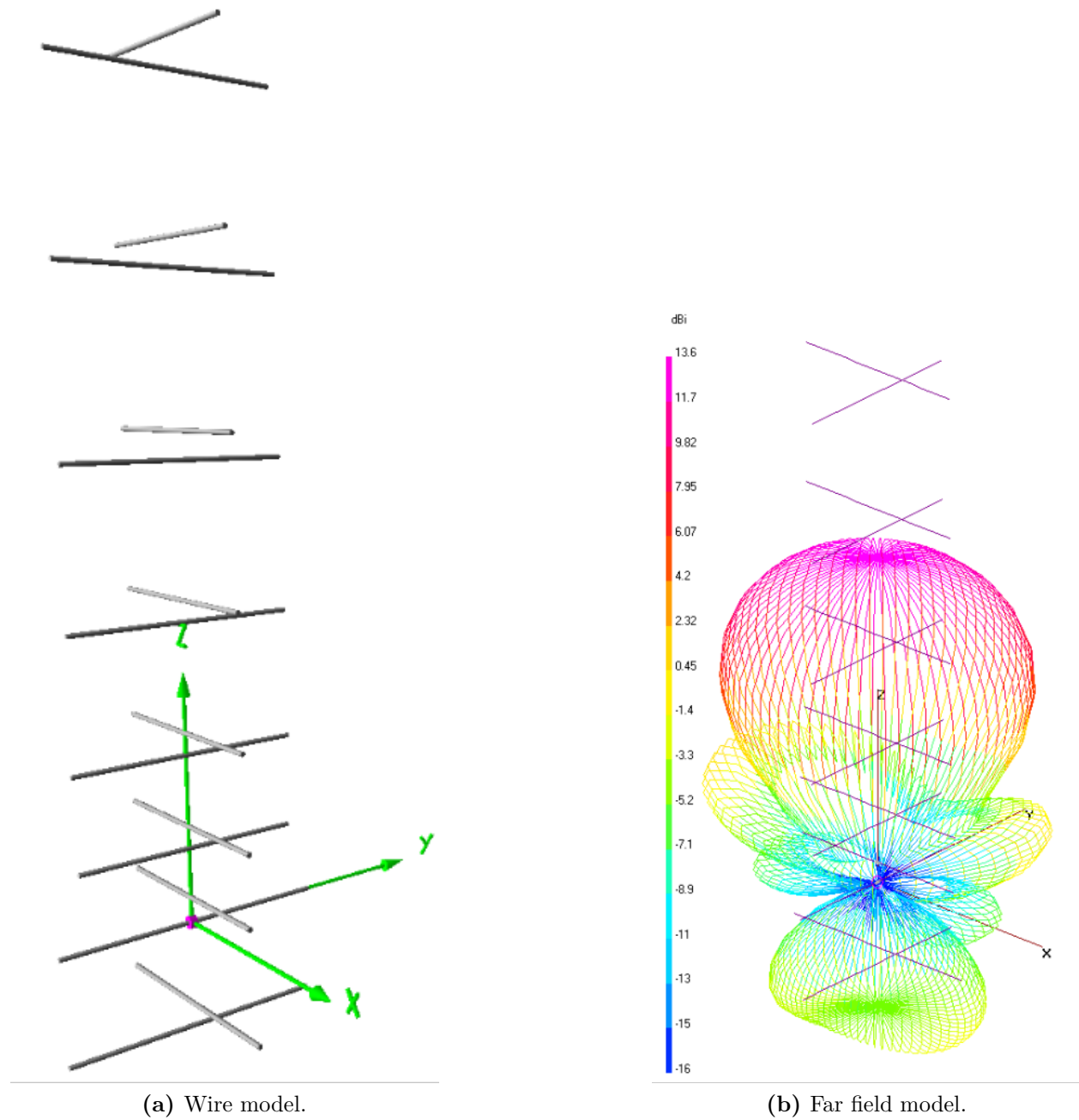
The 216CP-SatComMod is a 16 element Yagi-Uda design for the 2 m band. It has a single feed point to one element and can achieve a gain of 13.8 dB at the centre of its main lobe. This antenna is a modification of the 216CP two meter band experimental design model that is included with 4nec2. The modification involved adding a second set of orthogonal elements at an offset to enable differing polarizations within interference between the elements.

This antenna model shows how tracking antennas, which follows ExampleSat during its orbital pass, use two sets of elements to create circularly polarized antenna beams. The circular polarization (right or left handed turning are possible) enables the ground station to receive signal despite the fact that ExampleSat may not have its antenna orientated in a matching polarization.

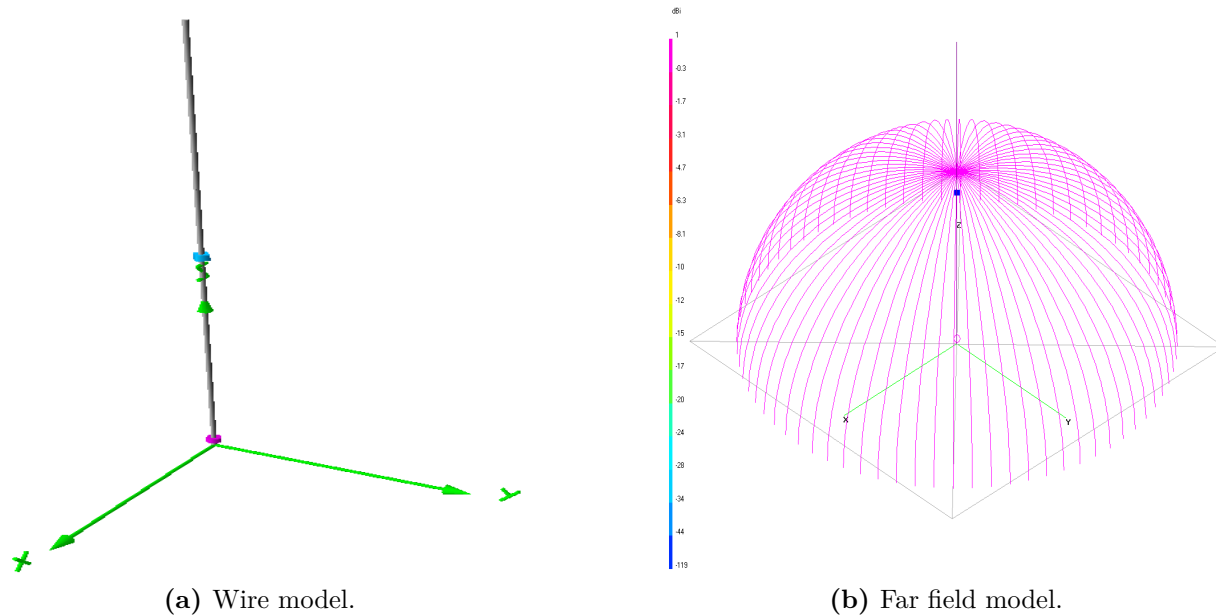
#### 4.4.2.2 Fixed I-Gate Ground Station Antennas

To simulate the typical I-Gate ground stations three different antenna models were created as described in Table 4.4 derived from survey data as shown in Table B.5.

The GP144-435 antenna design is a classic dual band (144 MHz and 440 MHz bands) "whip" antenna that uses the surface its mounted on as part of the radiating ground plane. Typically this design of antenna is attached to metal body of a car, a large metal plate or the ground surface itself. The ubiquity of this antenna design is due to its simplicity. Its possible to build this antenna out of any thin piece of wire and an AM/FM band form of this design was attached to most cars prior to 2000. Variations on this design are to include a magnet as part of the base, thus allowing it to be a temporary installation on the roof of a car. Its likely that this design would be most popular with the amateur radio community if asked about mobile operations. However, our interest in this research was related to the fixed I-Gate stations, hence the importance of the community survey in alleviating model selection design biases.

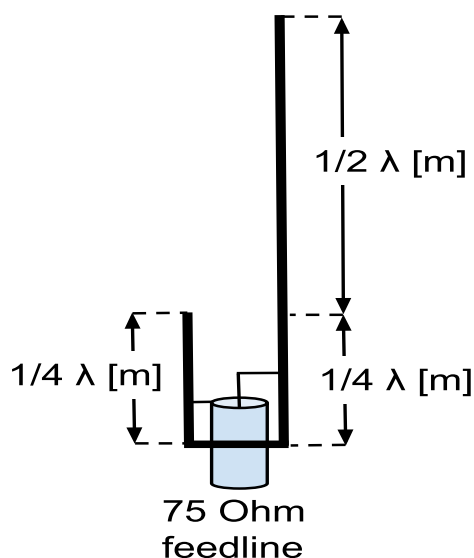


**Fig. 4.7:** VE4UM directional station antenna model in 4nec2.

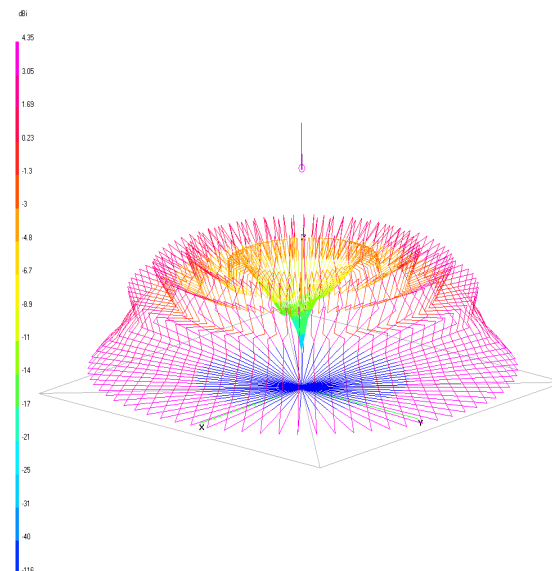


**Fig. 4.8:** GP144-435 omnidirectional antenna model in 4nec2.

J-Ant144, shown in Fig. 4.9, is a classic J-Pole design (also known as a Zepp' antenna), with a single  $1/2\lambda$  and two  $1/4\lambda$  elements fed by coaxial cable feed line from the bottom. In many designs of this antenna the radiative elements are contained in some form of plastic or PVC. The far field radiation pattern gives a "jagged layered bunt cake" shape.



(a) Wire model.

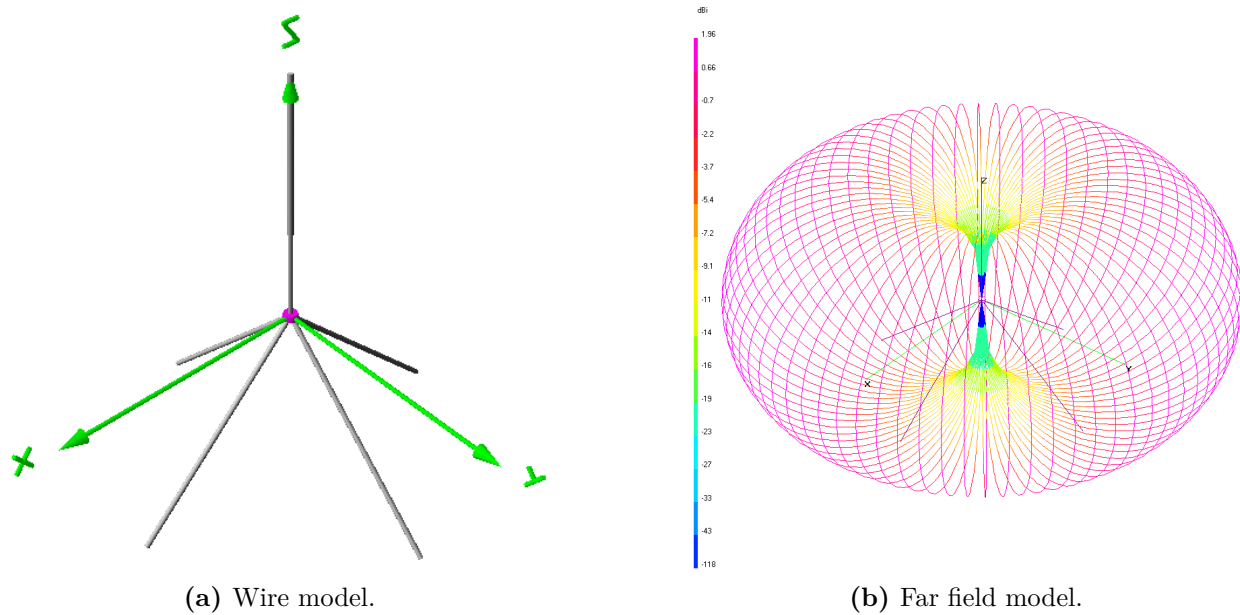


(b) Far field model.

**Fig. 4.9:** J-Ant144 omnidirectional antenna model in 4nec2.

The 2mGP antenna is a classic ground plane design. It has four (4) elements coming from the bottom of it to form the ground plane. This gives the far field radiation pattern a large apple shape as shown in Fig. 2.15b. As a surprise to the author, this was the most popular of the antenna designs by a large margin. It was initially assumed that the two (2) meter band J-pole as shown in Fig. 4.9 would be more popular as its easy to purchase or build as well as taking up less physical volume and is easier to mount. However, as we can see by the far field radiation pattern in Fig. 4.9b it does not project the idealized omnidirectional pattern. This further confirms the need for a more comprehensive and rigorous survey of the amateur radio community's antennas.



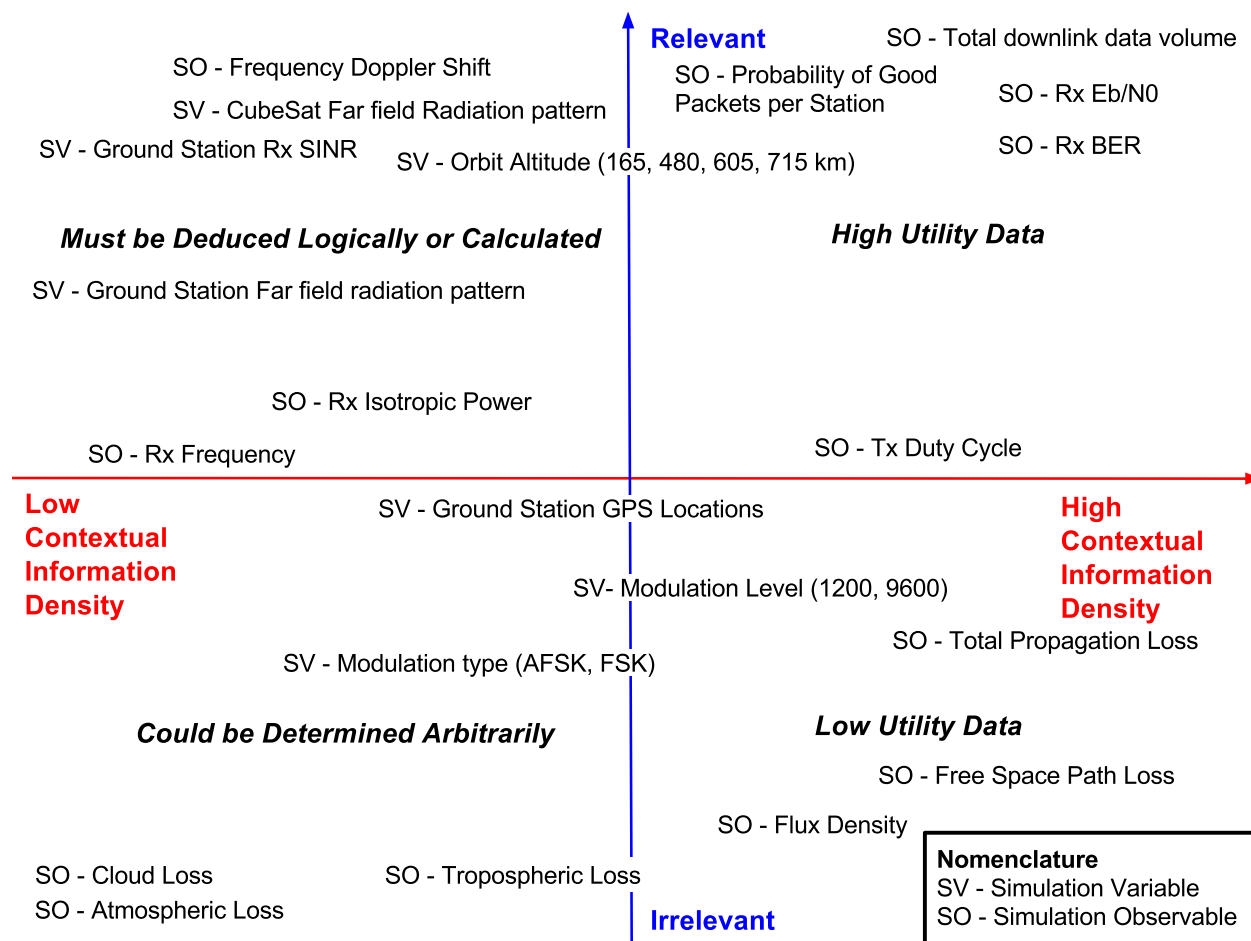


**Fig. 4.10:** 2mGP omnidirectional antenna model in 4nec2.

### 4.4.3 STK Variables and Observables

Within the STK synthetic environment a number of different variables that could be manipulated as well as a number of observables that could be monitored as a result. We must consider what variables are relevant to the simulation objectives as well as which observables will serve a purpose within the post collection analysis. To avoid “paralysis by analysis” in which we measure everything possible but gain little insight we can label the available variables and observables across a qualitative scale. To begin we choose two different axes. The vertical is a relative measure of the variable’s/observable’s level of relevance to the overarching thesis questions. The horizontal axis is a relative measure of the variable’s/observable’s contextual information density. This information density represents how much information within a specified context can be gleaned from this variable/observable. This contextual information density becomes biased towards observables or calculated values with a larger number of input variants from the simulation. We see in Fig. 4.11 variables and observables within the STK simulation can be aligned and create four descriptive quadrants.

The least relevant (irrelevant) and lowest contextual information density values occupy the lower left quadrant. These are calculated values that in and of themselves are necessary for typical link budget analysis but actually could be determined arbitrarily in our particular scenario. This is true of the tropospheric, atmospheric and cloud losses, all of which have similar traits. They all need to be accounted for in validatable simulations but could be arbitrarily determined or equated to zero as described in Sec. 4.3.1. Remember that for the frequencies of interest, 144.39 MHz or 437.385 MHz, the atmospheric losses according to the ITU propagation models can be considered negligible [ITU2013] since the frequency is under 10 GHz. Such assumptions are especially true when conducting trade-off studies since



**Fig. 4.11:** Diagram of possible variables and observables from STK

the channel being used is not considered fast fading with frequency within these bands. Thus these losses are insensitive to differing data rates, modulation types or frequencies. This implies that these observables are more irrelevant and have low contextual information density.

When Free Space Path Loss this observable is often highly requested calculation when determining link budget calculations. In actuality this observable represents a threshold value for first order magnitude calculations for a communications system designer. While important for further calculations this low utility observable in and of itself does not allow for much further study or improvement upon the design, beyond ideas such as changing the output power, moving one of the transmitter / receiver pair or selecting a different frequency.

Comparing the Free Space Path Loss to observables such as  $E_b/N_0$  we can see how much context and relevance such a calculation has. It includes the modulation type, the complete gain vs loss chain, and the channel noise. For the case of our simulations the gain vs loss chain also includes the variations on antenna design and orientation. The end goal of any digital communications design is to increase the overall utility by improving the signal to noise ratio along with the applicable data rate. Observables such as  $E_b/N_0$  afford more insight into trade-off study compared to Free Space Path Loss. Based on what we see in the

top right quadrant we will focus our efforts on the received  $E_b/N_0$  from ExampleSat-165 to any ground stations or I-Gates of interest. By relation we will also investigate the Bit Error Rate, packet reception and total downlink data volume.

By placing each of the available variables and observables within the relevance vs contextual information density graph we can focus our efforts on variables and observables that are more impactful to our research. Such a diagram could also be populated with Emulation Variables and Emulation Observables from the experimental design described in Sec. 3.4.

## 4.5 Verification and Validation

When using modeling and simulations there are two concepts that guide a researcher towards producing an accurate and credible model. Verification and Validation. These principles were best summarized by Boehm in his 1998 paper “Software Risk Management” for the European Software Engineering Conference [Boehm1989]. Verification: Are we building the product right? Validation: Are we building the right product?

### 4.5.1 Model & Simulation Verification

Verification is the process of confirming that the model is correctly implemented with respect to the initial concept. For this thesis verification was an iterative process in which the model is tested to find and fix errors within its implementation. Note that this loose definition allows for a continuum of realism based on the purpose of the model.

During the development of the STK simulations for this thesis there were 18 different versions of the same scenario. Some simulations were exploratory to see what could be modeled and what objects within the simulation required more refinement. Various components of models or objects were added in stages to ensure that the simulation fulfilled the needs of measuring the received signals from ExampleSat-165 to the nodes of the virtualized ground station network. One the simulation was for the purpose of generating uncluttered graphics showing the overall objective of the simulation as shown in Fig. 3.3. Several versions were made to test the custom Perl script used to import the I-Gate locations.

The far field antenna models created in 4nec2 were verified against available 2D plots when similar designs were available on the consumer market. This was done simply by confirming the visual shape of the far field model corresponded with the expected radiation pattern for a given wire pattern. This method of comparison is how the discrepancies between STK’s and 4nec2’s coordinate systems were identified and mitigated as discussed in Sec. 4.4.

### 4.5.2 Model & Simulation Validation

Validation is the process of confirming that the model or simulation represents the real system under study based on the users requirements. For this thesis validation was done using face validity, comparison to other models/simulations and by ensuring that all of the components in the models or scenarios were backed by citable sources. At face value the set of models and the scenario appear to behave as expected. Friends and colleagues were asked about the shape of far field antenna patterns as a sanity check for some of the results.

Unfortunately there is no practical way to test the simulation against historical data since there is no North American wide terrestrial based satellite reception network that is not proprietary in nature. A practical experiment is not possible without creating and launching a CubeSat itself. This process typically takes several years to design and build with up to four years to negotiate the downlink frequency via the ITU (section 3.7.3 CanX-2 of [Klofas2008]).

From a usability standpoint the simulation includes enough components to provide useful data to evaluate the link budgets and the network designs as required. Several objects were placed into the scenario to allow for further comparison of different orbital altitudes. The simulation is flexible enough to iterate for further research.

#### 4.5.2.1 Limitations of the Models or Simulations

One key difficulty of the model of this scope is the practicality of calculations involving multipath. While in the VHF or UHF domain calculations related to multipath fading in an idealized uncluttered WGS84 oblate spheroid are possible with a single ground station, such a simulation is not practical when adding 842 different nodes over as wide an area as North America. Adding in urban terrain features such as buildings and other sources of reflection other than the Earth's surface to such a detailed model could prove difficult for a single computer to calculate within STK.

A second key limitation with the models is the number of ground station far field antenna models used. Because of the limited survey respondents from the amateur radio community the data would not be a statistically valid representation of the real population of antennas used.

Within the STK simulation tool itself optimization is not practical without a very manual and iterative exploration of the design space. Add on utilities such as STK Analyzer [AGI2016b] and its extension STK Optimizer [AGI2016c], both created by third party developer Phoenix Integration Inc., allow for the automation of analysis. Analyzer includes the capabilities for parametric studies, carpet plots, design of experiment algorithms, Monte Carlo inputs, and trade studies. Optimizer, which extends the capabilities of Analyzer, includes the ability for gradient-based, adaptive surrogate, and genetic algorithm based optimization. These automation tools were purchased and explored for utilization within this thesis. These tools for analysis can make quick work of comparing between different variables or options within a scenario. However, there was little to no support available from Phoenix Integration Inc. on what the underlying functions were, how calculations were made nor why certain design choices were made within the software add on. From a day to day engineering perspective Analyzer and Optimizer could prove to be valuable tools, simplifying and expediting a design analysis. From a research perspective these tools should be considered black boxes that simply provide an output without much documentation on how they work or why. Because of this lack of documentation or support for research these add-ons could not meet the requirements of verification nor validation. For this reason Analyzer and Optimizer were not used to generate nor study any of the data within this thesis.

## 4.6 Summary

In Sec. 2.5 we saw how a literature review for the population of CubeSats led to a suite of conceptualized ExampleSat models. The worst case scenario (from a total access time perspective) where the satellite is in the lowest commonly used orbit of 165 km was chosen to create the more specific ExampleSat-165 conceptual model. Further attributes to this model included its specific orbit and communications abilities including its antenna design. In Chapter 4 we expanded upon ExampleSat-165 by using 4nec2 to create a far field radiation pattern for its transmitting antenna. From here the pattern was imported into STK and attached to the ExampleSat-165 object as an external antenna pattern. Combined with the communications settings as described in Sec. 4.3.2 this creates a generalized model for a typical CubeSat at a commonly occupied orbit. To model the virtualized ground station network custom Perl code was used to extract the I-Gate call signs and locations. These I-Gates were also imported into STK where communications links were modeled between the virtualized ground station network nodes and ExampleSat-165. Link budgets and access times were then generated as simulation observables for analysis in Chapter 5.



# 5 Experimental Results and Discussion

## 5.1 Overview

As we saw in Chapter 4 and in particular Sec. 4.4.3 the possible observables from the simulation are numerous but we can narrow down our area of interest to values such as the revved energy per bit relative to noise,  $E_b/N_0$ . Throughout this chapter we will explore and analyze the outputs from the simulation. We will begin by looking at what the STK simulation tool can tell us about the communications link. Based on this data we will then characterize this communications link in particular how the packets are received. From that point we move onto a statistical analysis of the received signal and compare two techniques of classifying the received transmission's stationarity. For completeness we will also include a short discussion on Doppler Shift and Doppler Rate as they related to the proposed emulation experiment from Sec. 3.4.

## 5.2 STK Outputs

STK is a very flexible modelling tool. It has prebuilt lists of output reports based on the typical observables of interest as well as the ability to create customized calculations. For example, once the object model ExampleSat-165 is populated within the synthetic environment a number of analysis options are available such as calculations based on the vector geometry of the satellite to any other object or frame of reference in the simulation. In the vernacular of STK reports can be based on “Data Providers” of most often based on vector geometry, transmitters or receivers. It is possible to create other calculations based on any other dependent or independent calculation. For the purposes of this research two types of predefined reports and one customized report were used. The first predefined report is a is an access summary report as displayed in Table 2.1 and compiled in aggregate in Table 5.3. Shown in Table 5.1 is an example summary of the access times when ExampleSat-165 has line of sight (LOS) access to VE4UM's I-Gate node for a period of 24 hours starting from 21 Oct 2015 05:00 UTCG.

This table tells us the total amount of time we have to communicate from ExampleSat-165 to VE4UM's I-Gate station. The table also helps with “sanity checks” of or simulation. If the access times result in something well outside the expected values we know to double check the settings of the objects within the scenario.

The second predefined report is a link budget. The simplified output form includes; Time, the Effective Isotropic Radiated Power (EIRP) ( $dBW$ ), Received Frequency ( $Hz$ ), Received. Isotropic Power ( $dBW$ ), Flux density ( $dBW/m^2$ ), antenna gain to noise temperature  $g/T$  ( $dB/K$ ), analog Carrier to Noise ratio by Hertz  $C/N_0$  ( $dB * Hz$ ), Bandwidth ( $kHz$ )  $C/N$

**Table 5.1:** Access Summary Report from ExampleSat-165 to VE4UM I-Gate.

Access	Start Time	Stop Time	Duration (s)
1	21 Oct 2015 05:30:23.308	21 Oct 2015 05:33:38.127	194.849
2	21 Oct 2015 06:56:12.712	21 Oct 2015 06:56:50.077	37.5
...	...	...	...
10	21 Oct 2015 18:24:52.687	21 Oct 2015 18:25:44.884	52.197

Min Duration	36.302
Max Duration	280.406
Mean Duration	113.072
Total Duration	1130.716

(dB) , Energy per Bit relative to Noise  $E_b/N_0$ (dB) and Bit Error Rate (BER). An abridged version of this report is shown in Table 5.2.

**Table 5.2:** Link Budget Report from ExampleSat-165 to VE4UM I-Gate (1 s interval).

Time	EIRP (dBW)	Rcvd. Freq. (Hz)	...	$E_b/N_0$ (dB)	BER
21 Oct 2015 05:30:23.308	1.633	144391830.164464	...	-6.6664	$3.212625 \times 10^{-1}$
21 Oct 2015 05:30:24.000	1.658	144391820.744011	...	-0.0810	$1.609111 \times 10^{-1}$
...	...	...	...	...	...
21 Oct 2015 16:53:45.100	7.571	144393195.287505	...	76.2440	$1.000000 \times 10^{-30}$

For the purposes of the analysis  $E_b/N_0$  is the significant measurement for the chosen downlinks based on the discussion in Sec.4.4.3. For our exploratory data analysis we first investigate the received  $E_b/N_0$  from ExampleSat-165 to a traditional AMSAT/OSCAR station using a Yagi-Uda antenna as shown in Fig.4.7. We can see in Fig.5.1 that there are only two access periods in which ExampleSat-165 has LOS with ground station despite the fact that it is using a directional antenna with a rotor (see Fig.2.7).

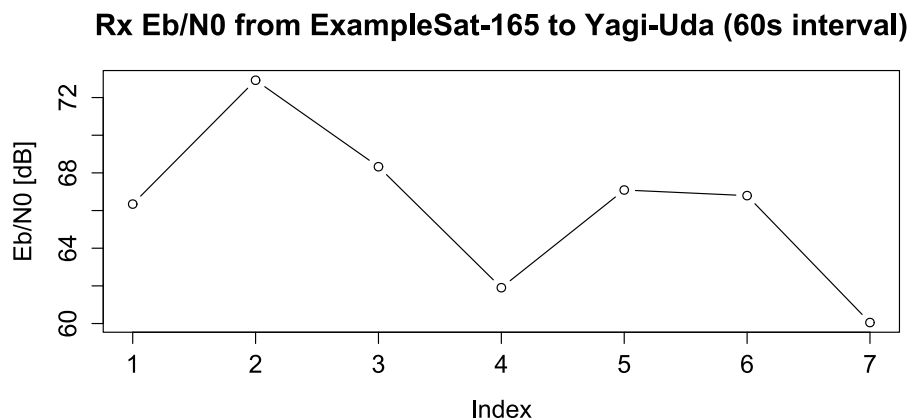
To make sure that we are actually looking at two access durations and not some artifact of the calculation interval of 60 seconds, we run the calculation again with a one (1) second interval as shown in Fig.5.2.

For comparison we can look at the received  $E_b/N_0$  from ExampleSat-165 to VE4UM's APRS I-Gate station that uses the 2mGP antenna shown in Fig.4.10. We can see in Fig.5.3 that four (4) different access periods are visible.

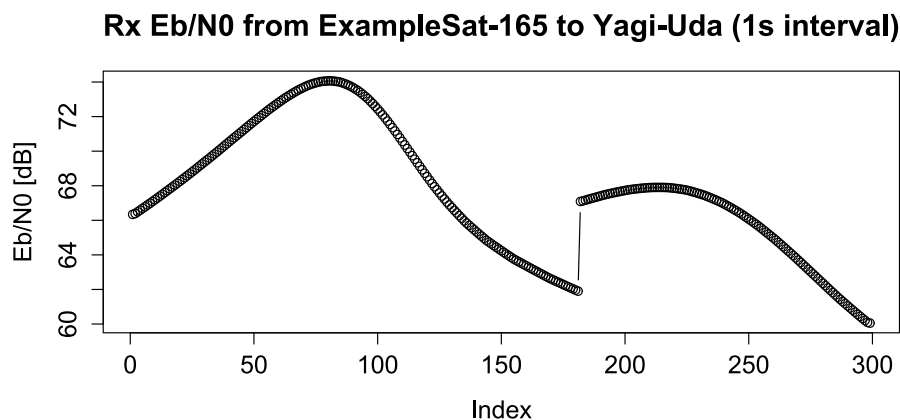
The calculation interval is narrowed to one (1) second in Fig.5.4, which has a smoothing effect across the four access intervals.

At first glance this appears counter intuitive. But as we tabulate the access times we can alter the minimum elevation angle ( $\theta_{min}$ ) of the Yagi-Uda antenna rotor within the simulation. Under typical operating conditions  $\theta_{min}$  would be set to a minimum of ten (10) degrees above the horizon. This minimizes the impact of terrestrial noise sources or multipath interference due to reflections from the ground or nearby buildings. However, since the I-Gate nodes use fixed omnidirectional antennas as discussed in Sec.4.4.2.2 its not realistic nor useful to include a  $\theta_{min}$  limitation on I-Gate nodes. As such the VE4UM





**Fig. 5.1:** Received  $E_b/N_0$  from ExampleSat-165 to VE4UM AMSAT/OSCAR Yagi-Uda antenna (60 s interval).

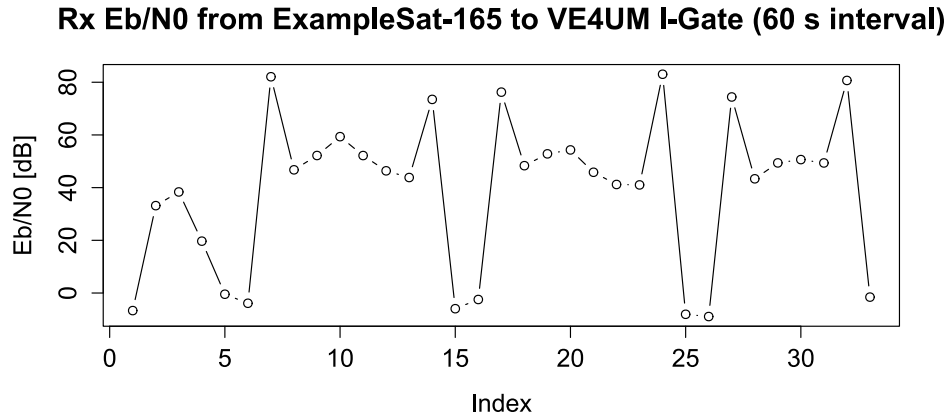


**Fig. 5.2:** Received  $E_b/N_0$  from ExampleSat-165 to VE4UM AMSAT/OSCAR Yagi-Uda antenna (1 s interval).

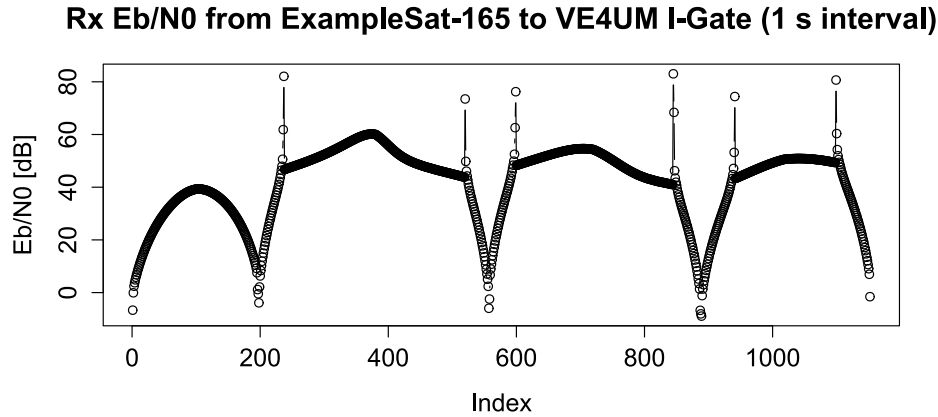
I-Gate node is only limited by LOS over the Earth's surface and therefor has more available accesses by comparison. As shown in Table 5.3 the access time for the Yagi-Uda station is comparable if the minimum elevation angle is disabled. Note that the Total Access Duration does account for overlaps in access time and is not a simple summation.

We must not conflate this concept as the need to turn off the minimum elevation angle settings for conventional AMSAT/OSCAR radio antenna systems as terrestrial noise sources or interference are not a part of this discrete event simulation. It would not be practical to attempt the multipath calculations or distributed interference sources necessary for a North America wide area of interest in a single PC. This limits the possible fidelity of the simulation.

Recall that the total access times account only for the duration of LOS with ExampleSat-165. It does not account for the time in which ExampleSat-165 has a clear LOS but the received  $E_b/N_0$  is low enough that the BER creates too many errors to be interpretable



**Fig. 5.3:** Received  $E_b/N_0$  from ExampleSat-165 to VE4UM I-Gate using 2mGP (60 s interval).



**Fig. 5.4:** Received  $E_b/N_0$  from ExampleSat-165 to VE4UM I-Gate using 2mGP (1 s interval).

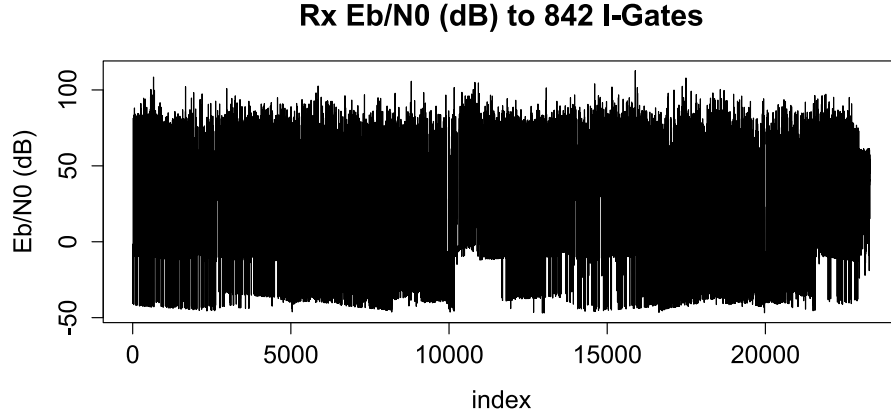
within the packet stream.

Plotting the received  $E_b/N_0$  for the entire virtualized ground station network of 842 I-Gates, as shown in Fig. 5.5, we see that the full series appears to have a similar distribution as White Noise. A short sample of the first 500 data points is displayed in Fig. 5.6 to provide clarity. Note that this time series represents all of the received  $E_b/N_0$  values without any figure of merit limitations such as only including data points that would result in correctly interpret bits or packets. We can consider the  $E_b/N_0$  values in within Fig. 5.5 to represent “nearly raw” data because any stationarity analysis must begin with as complete a time series as possible.

We must note that the data shown in Fig. 5.5 and all subsequent analysis has erroneous values removed. Due to the nature of simulated environments there is a calculation error within the raw data set as the null region of ExampleSat-165’s far field radiation pattern aligns with the null region of a ground station node’s corresponding far field radiation pattern.

**Table 5.3:** Access Times comparisons.

Ground Station	Access Duration [s]				Accesses
	Min	Max	Mean	Total	
VE4UM I-Gate 2mGP	36.302	280.406	113.072	1130.716	10
Yagi-Uda ( $\theta_{min} = 10^\circ$ )	116.211	179.454	147.832	295.664	2
Yagi-Uda ( $\theta_{min}$ disabled)	36.341	279.973	113.246	1132.456	10
Virtualized Network	42.066	1092.960	345.926	10292.846	23k+

**Fig. 5.5:** Received  $E_b/N_0$  from ExampleSat to all North American I-Gates.

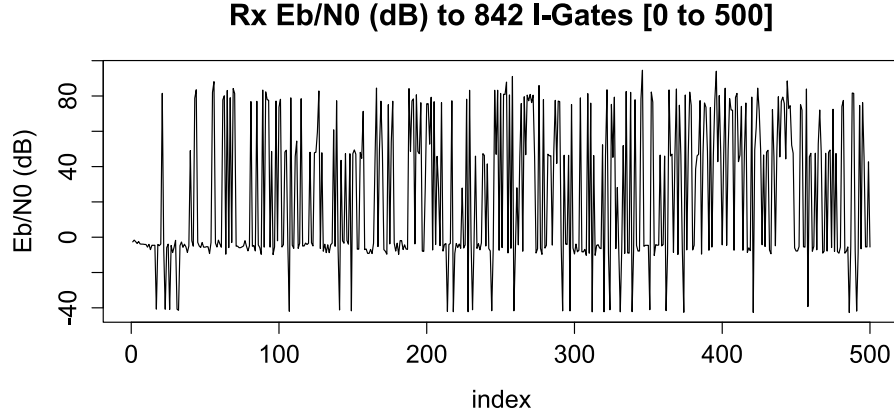
The result is an  $E_b/N_0$  in excess of 3000 dB, is a logical impossibility given the limited transmission power of the satellites and the relatively low gain of both antennas. Such  $E_b/N_0$  values within the data set were anomalous based on manually observing the times when they occurred within the simulation environment. These anomalous values were deleted from the data series as a set once they were demonstrated to be statistically insignificant at the five percent (5%) level via simple ANOVA.

## 5.3 Packet Reception

Now that we have the received  $E_b/N_0$  values and their associated BER values we have the opportunity to determine the behavior of the network from the perspective of the received packets themselves. As we discussed from the results of Table 5.3 we must consider how packets are received and their likelihood of reception based on the received  $E_b/N_0$ .

We know the received  $E_b/N_0$  and BER from the simulation observables. For example with a received  $E_b/N_0$  of -11.9998 dB the BER is  $4.01 \times 10^{-1}$  or 40.10% representing the probability that a bit is received in error. From this point we can determine the Probability of Packet Reception,  $P_{PacketRx}$  using the value of BER based on the previously mentioned (Sec. 2.6.1) works of Goode and Wallio [Goode1984, Wallio2015].

$$P_{PacketRx} = (1 - BER)^{N_p} \quad (5.1)$$



**Fig. 5.6:** The first 500 samples from Fig. 5.5.

Where  $N_p$  is the packet size of 2600 bits which is the maximum size of an APRS packet without a destination address as described in Sec. 2.4.3. By calculating the  $P_{PacketRx}$  for each output value of BER provided from the link budget report, as described in Table 5.2, and the associated  $E_b/N_0$  we can show the relationship between the received energy from the spacecraft and the likelihood of a given packet being interpreted correctly. If we edit the received  $E_b/N_0$  data set to include  $P_{PacketRx}$  only values between 0.00% and 99.999% we can create a graph as shown in Fig. 5.7. This editing of the data set is necessary due to the large number of points that are at the extremes of 0% and 100%, making it difficult to view the relationship. For analysis beyond the limited scope of packet reception such data points will remain in the set.

The data in Fig. 5.7 was fitted to a hyperbolic tangent curve.

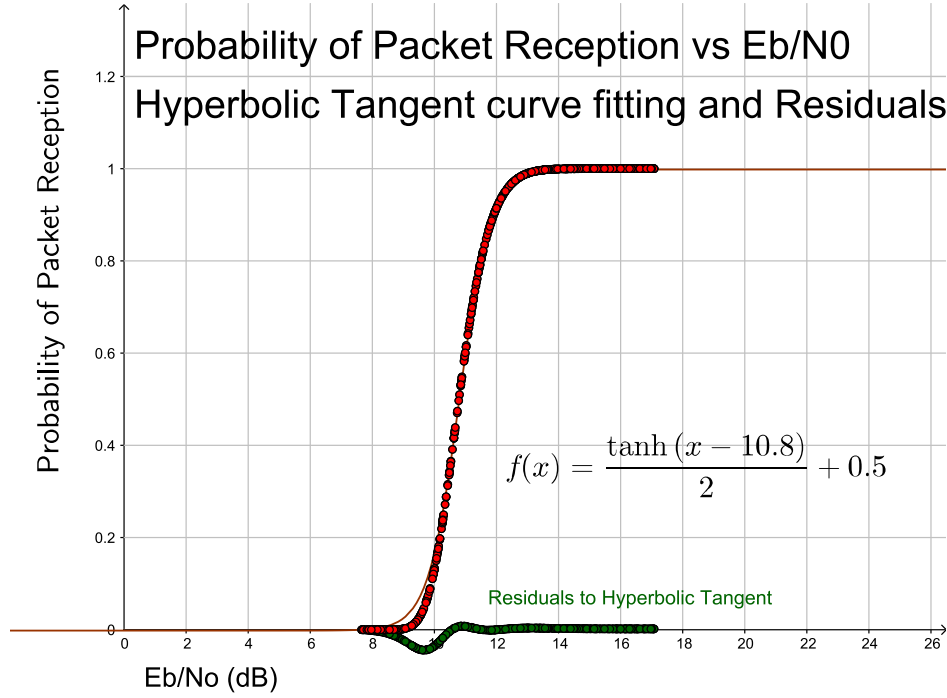
$$f(x) = \frac{\tanh(x - 10.8)}{2} + 0.5 \quad (5.2)$$

The plot also shows the residuals for a graphical representation of the fit of the hyperbolic tangent to the data set. We can from the residuals in the plot that fit for the hyperbolic tangent is less than ideal when  $E_b/N_0$  is 10 dB or less, however this would occur when the Probability of Packet Reception is 20% or less and is likely of little practical impact as a result.

Using Eq. 5.2 enables us to predict the minimum  $E_b/N_0$  for a required probability of packet reception, for values between 0 and 1, that includes the complete link budget chain from ExampleSat-165 to the virtualized ground station network.

$$E_b/N_0 = \frac{54}{5} - \tanh^{-1}(1 - 2P_{PacketRx}) \quad (5.3)$$

The equations presented above demonstrate the utility of creating a simulation based on object models that include the complete link chain such as far field antenna patterns, the satellite's orbit, real world GPS locations and ITU propagation models. This methodology



**Fig. 5.7:** Plot of Probability of Packet reception vs Received  $E_b/N_0$ .

could prove to be a powerful design tool as it enables design trade studies to evolve beyond conventional link budget analysis but to the optimization of correctly interpreted packets or indeed the total data volume.

We can also look at the relationship directly between the  $P_{PacketRx}$  and the BER. As shown in Fig. 5.8 as the BER increases the likelihood of interpreting the packet correctly drops.

Most importantly we can see how  $P_{PacketRx}$  and BER relate through an exponential curve with a coefficient of determination of one (1).

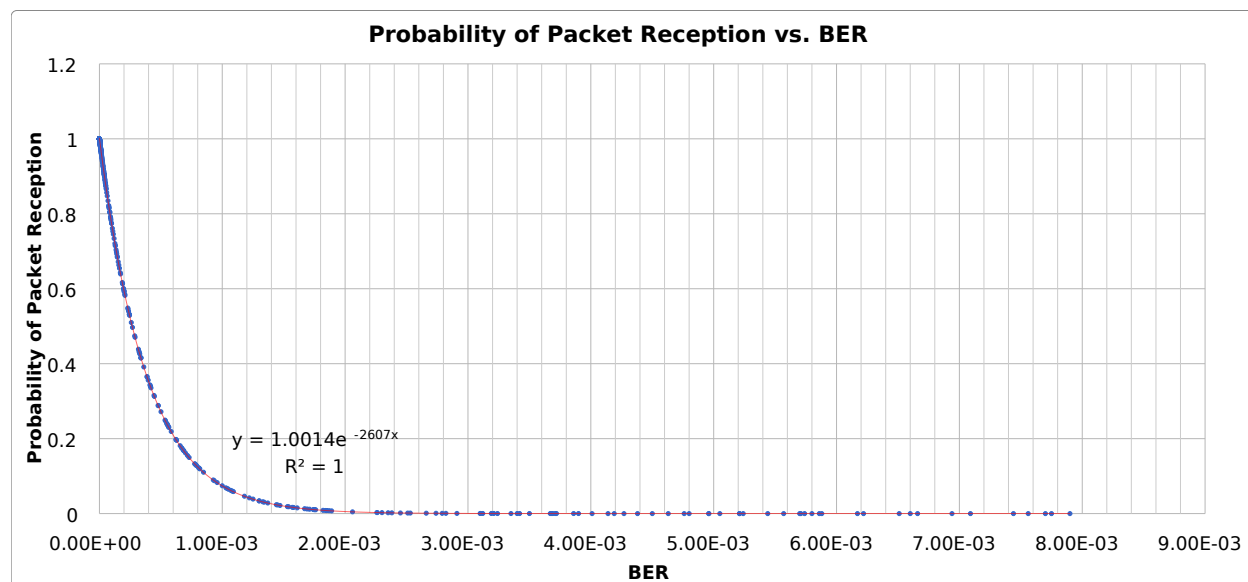
$$P_{PacketRx} = 1.0014e^{-2607(BER)} \quad (5.4)$$

In the interest of completeness for the set of design tools we can also examine the relationship between the BER and the  $E_b/N_0$  as shown in Fig. 5.9 with the residuals as well.

$$BER = e^{-[(E_b/N_0) - 2.93]} \quad (5.5)$$

This plot provides us with a convenient tool for determining the required  $E_b/N_0$  to achieve a desired BER through Eq. 5.5 based around the virtualized ground station network. The set of equations presented above characterize the ability for any ground station across the virtualized ground station network to correctly receive an APRS packet from ExampleSat-165. We must keep in mind that this represents the interpretation of a single packet, whereas ExampleSat-165 will be transmitting a stream of packets to the ground station nodes.

Based on the AX.25 UI Packet Frame as shown in Table 2.3 we can create another table that shows the relationship between the scientific data payload size and the total packet



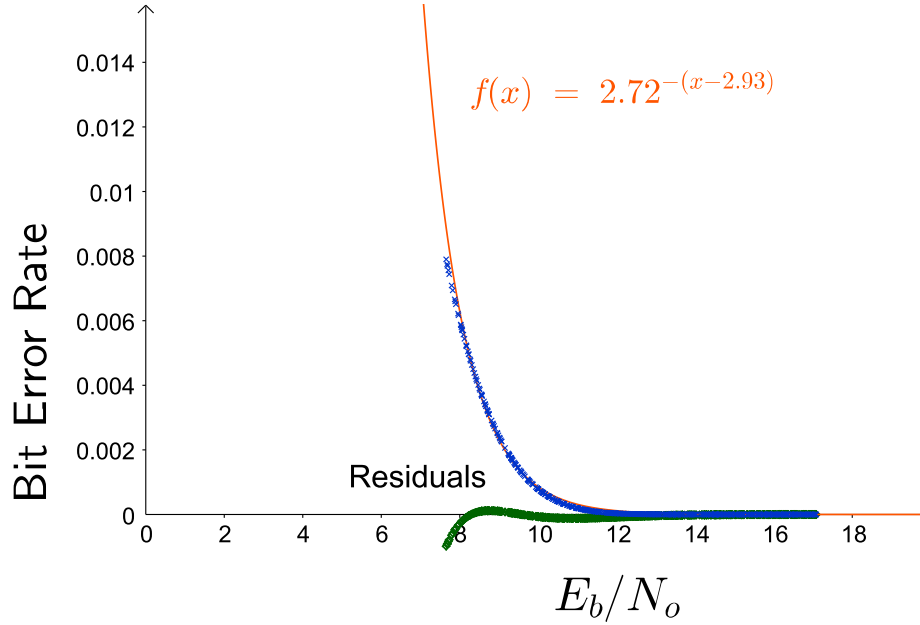
**Fig. 5.8:** Plot of Probability of Packet reception vs Received BER.

volume with its associated overhead assuming a maximum user data volume of 256 bytes. Remember that AX.25 does not natively include any form of Forward Error Correction (FEC) so such measures must be included within the Science Data field of the packet otherwise the network will not propagate the packet correctly. Each field within the AX.25 packet is proscribed for a set network routing purpose with the exception of the science data field itself.

**Table 5.4:** Science Data Payload to Total Packet Volume.

Science Payload (kB)	Packets	Total Payload (bits)	Total Packet Volume (bits)
1	32	8192	10400
2	63	16128	20475
...	...	...	...
2000	62500	16000000	20312500
5000	156250	40000000	50781250

Looking at Table 5.4 we can see how the overhead within the combined APRS and AX.25 protocol inflates the number of packets required and thus the total volume of bits. A 5000 kilobyte (kB) of 5 megabyte (MB) file would require 156250 packets for a total of 50781250 bits. While using the APRS required baud of 1200 this would take 42317.71 seconds (705.295 minutes or 11.755 hours). Based on the results of Table 5.3 this would take 4.1 days to download a 5 MB science payload by the virtual ground station network compared to 143.128 days for a single AMSAT/OSCAR Yagi-Uda equipped ground station. This represents a 97.14% decrease in the required downlink time. This assumes that all the packets were received correctly which is only possible if the  $E_b/N_0$  is 13 dB or greater based on Fig. 5.7. This leads us to a graphical analysis of  $P_{PacketRx}$  as the science payload data increases the number of required packets.



**Fig. 5.9:** Plot of Received BER vs Received  $E_b/N_o$  with residuals.

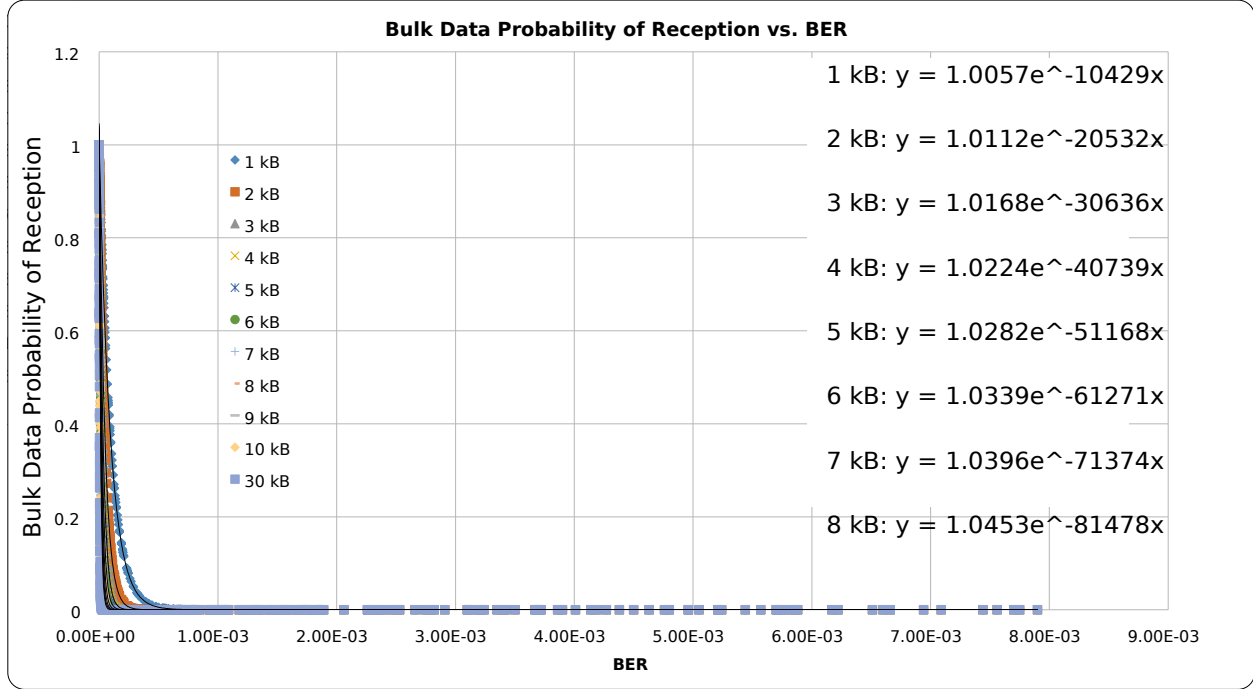
In Fig. 5.10 we can see the change of the exponential relationship between Bulk Data Probability of Reception and BER. As the total science payload data described in Table 5.4 increases the total packet volume causes an inflection in the exponential relationship such that after 8 kB in size it's no longer practical to use trend analysis as illustrated by Fig. 5.11. Note that all of the exponential trend lines presented in Fig. 5.10 have a coefficient of determination ( $R^2$ ) of one (1) indicating an ideal fit.

This figure demonstrates that utilizing AX.25 based APRS for larger data volumes requires on-board “bulk breaking” of the data set not only for reasons of the limited LOS from ExampleSat-165 to a ground station node, but also because of the limits of reception due to BER. Remember that although the above analysis looks at the data set of received  $E_b/N_o$  as if it were predictable, in and of itself, this fact remains to be determined. The preceding leads us to look at the time series data's stationarity in Sec. 5.4.

## 5.4 Non-Stationary Time Series Analysis

One proposed method of analysis is to sample the time series data across sequential short windows. The key notion is that over a selected window of time could the data series appear to be stationary? We will call this method Short Time Window Analysis of Non-Stationary Time Series. Three questions need to be addressed. Can we differentiate between stationary and non-stationary series with this technique? If so, does a change in short time window size have a statistically significant impact? If it does, what generalized method of window selection is appropriate for the time series for analysis?

We begin by collating the first moment; the mean value for each sequential window for a given window size. It's prudent of us to examine the first moment, the mean, of the



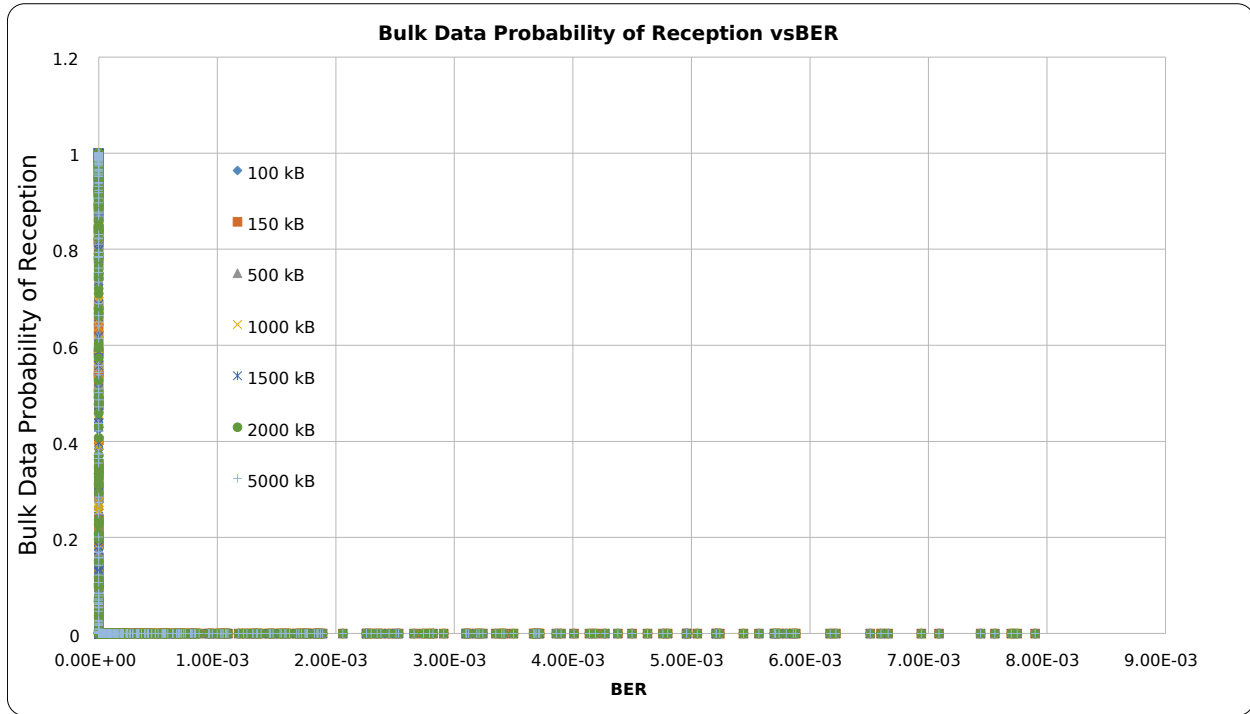
**Fig. 5.10:** Science Payload Data Probability of Packet Reception vs BER.

data set as a starting point to see if a strong sense stationarity detection is possible with this method. We have arbitrarily chosen window sizes from 50 to 750 with increments of 50. This sampling interval was selected as it provides a large enough sample of means to accommodate the central limit theorem, if applicable, for both the window itself as well as across the set of windows dividing the entire series of 23306 data points (32 windows of 750 data points per window). This differs from the principle of a moving average window in that the Short Time Windows do not overlap. Each window size divides up the entire data series into equal volume sets. By examining the mean value of these windows across time we may be able to quickly and easily determine if the series is stationary. This method could have a computational advantage as its simplistic to calculate and potentially perform comparisons with.

To see if any analysis technique that we concoct is of any use we must have some comparative process with known stationary and known non-stationary data. By definition the unpredictable randomness of White Noise acts as our know stationary process. Since Brown Noise is based on a random walk process, and thus each point is dependent on the last, by definition it is a known non-stationary process. For these reason we have generated two series of 23306 data points using both White Noise (Fig. 5.12a) and Brown Noise (Fig. 5.12b) as described in Sec. 2.6.2. The length of each series was to provide a practical scale relative to our received  $E_b/N_0$  time series data (Fig. Fig. 5.5) that was originally collected over a simulated 24 hour period.

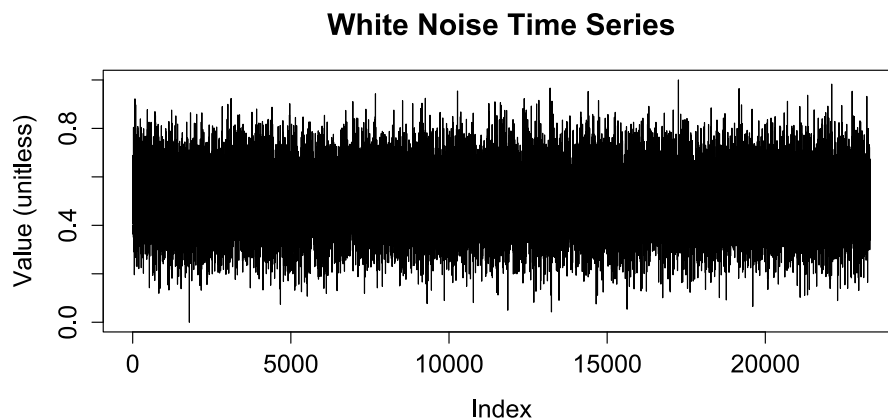
Each of the analysis techniques will be applied to all three sets of data to compare their efficacy with known stationary (White Noise), known non-stationary (Brown Noise) and unknown stationarity series ( $E_b/N_0$  time series). As mentioned in Sec. 2.6.2 the R programming language [Ihaka2016] was used to generate the White and Brown Noise data series.



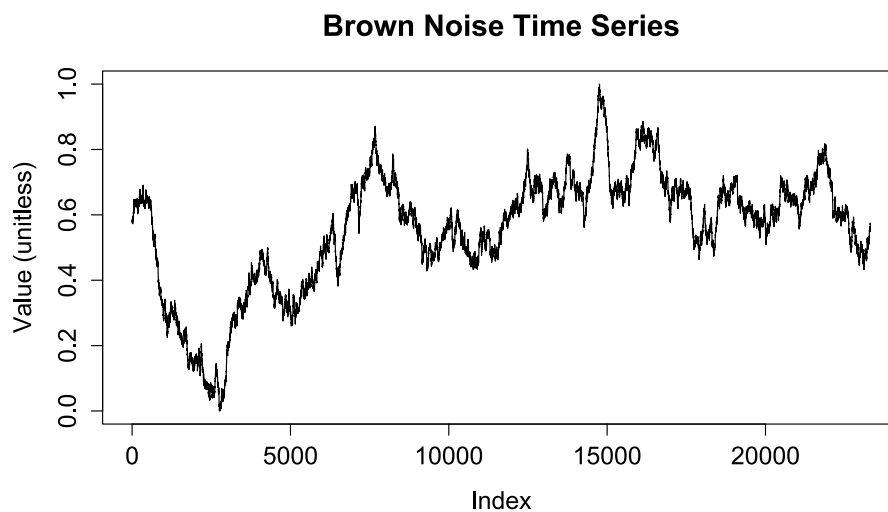


**Fig. 5.11:** Bulk Science Payload Data Probability of Packet Reception vs BER.

Specifically, each series was generated by the TK95 function from the {RobPer} package [Thieler2016]. This particular tool for noise generation has been cited in approximately 381 articles as of January 28<sup>th</sup>, 2016, many related to spectral analysis for stellar observation. Thus we assume that this algorithm for noise generation is reasonable for use in our radio frequency application.



(a) White Noise time series data.



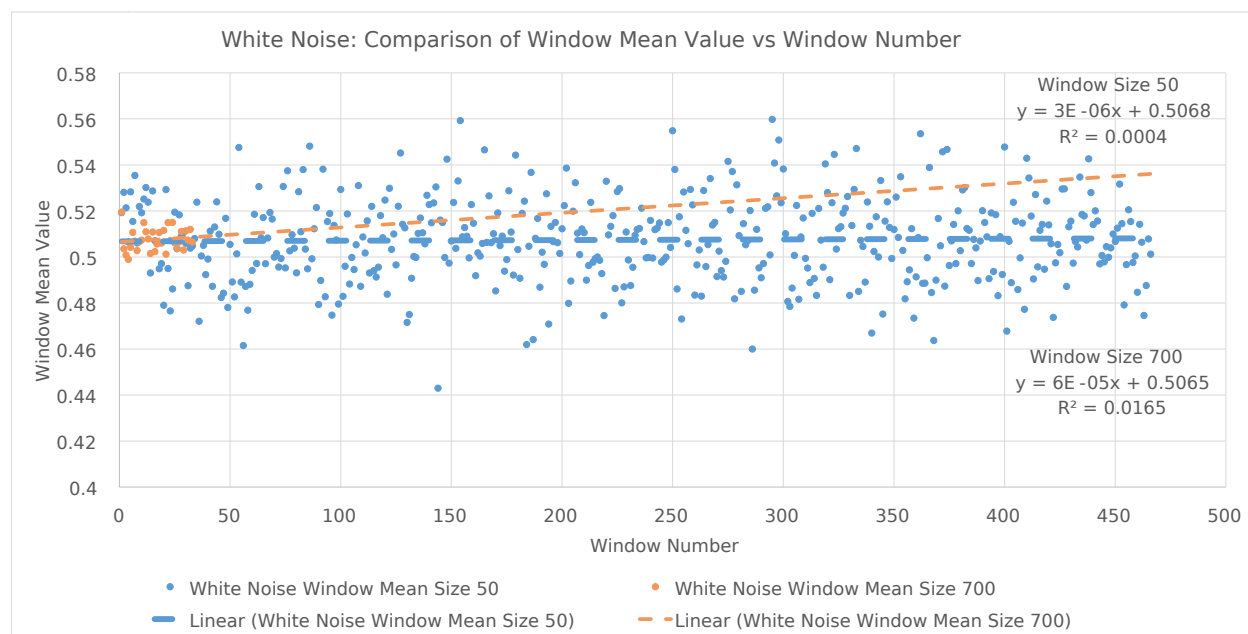
(b) Brown Noise time series data.

**Fig. 5.12:** White and Brown Noise time series data generated via power law.

### 5.4.1 Slope Comparison

The mean value for each window is calculated and then plotted in sequence. The slope of a linear best fit line for the set of short time windows indicates the rate of change of the mean across time. This becomes a graphical representation of the change of the mean over time.

Thus if the slope changes we could state therefore that the mean has changed and thus the series is not stationary. By definition we know that White Noise is generally considered strict sense stationary. However, we must note that this definition stationarity is actually within an order of variance from the grand mean for the series as we would with any statistical representation. As such the slope for any plot of White Noise we should also expect some variation in slope between the best fit lines of different Short Time Window. Of particular interest is to determine if there is significant change in slope for the Short Time Windows of White Noise as it would be an indicator of the level of utility for this method.

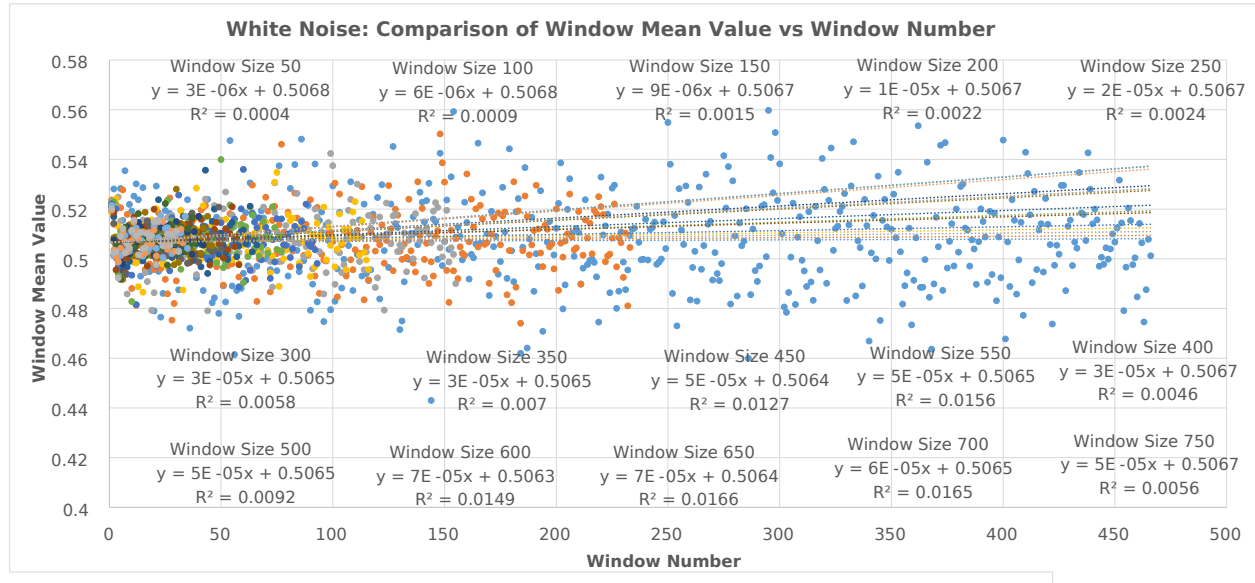


**Fig. 5.13:** Plot of White Noise short time window means vs window number with comparative slopes; window sizes 50 and 700 samples.

We can see in Fig. 5.13 a plot of short time window means vs window number for the White Noise data series. This particular plot shows only the data sets for windows sizes of 50 and 700 data points to illustrate the principle of slope comparison between window sizes. The plot area shows the equation for the best fit line of each data set and the associated slope. Also displayed is  $R^2$  as an indicator for how well the linear model fits the data set. Since we are using simple linear regression the notation of  $R^2$  is equivalent to the classical use of  $r^2$ , the difference in nomenclature is an artifact of the automatic notation in the plotting software used.

Based on the low values for  $R^2$  we can see that the linear best fit line is not an ideal model to represent each data set. The intention of our analysis is not to create an accurate model for each short time window size. Our goal is to compare the mean values of these

windows across time as well as across short time window sizes. In this way a simple linear best fit model provides a generalized relative measure for comparison without the need for individualized models per data series or per short time window size. By using a linear best fit trend line model we may be able to perform sensitivity analysis of stationarity based on data series and short time window size. Effectively the slope becomes a measure of tendency across time for differing window sizes. The value is easy to calculate and plot but can only be used as a comparative measure.



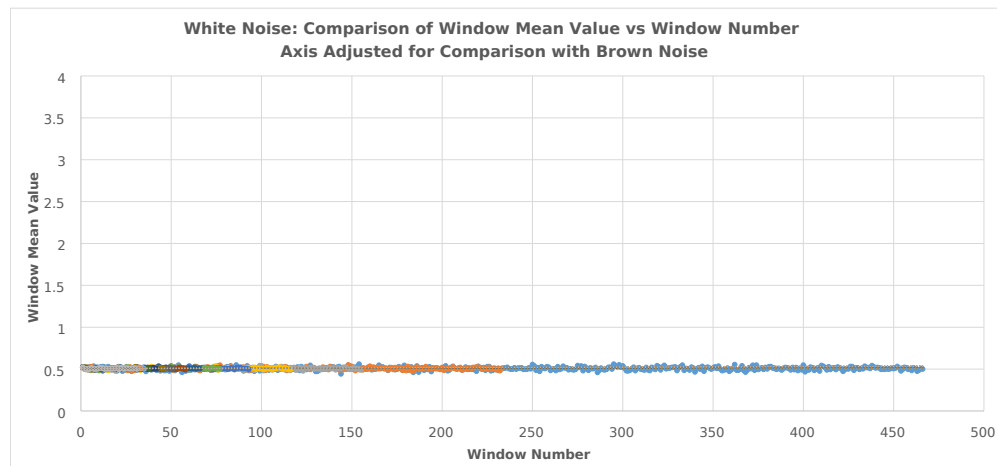
**Fig. 5.14:** Plot of White Noise short time window means vs window number with comparative slopes.

This principle becomes clear when looking at Fig. 5.14. The plot is similar to Fig. 5.13 but instead shows all of the means for the short time window sizes against the window number. In this way we can graphically see the level of heteroscedastic tendency present as the data is aggregated as the short time window size increases. Again all of the linear best fit model details regarding slope and coefficient of determination were collected. This process was repeated using White Noise (Fig. D.2), Brown Noise (Fig. D.3) and the received  $E_b/N_0$  from ExampleSat-165 (Fig. D.7) to all of the APRS I-Gates in North America as shown in Fig. 5.5. Each of these plots can be found in Sec. D.1.1.1.

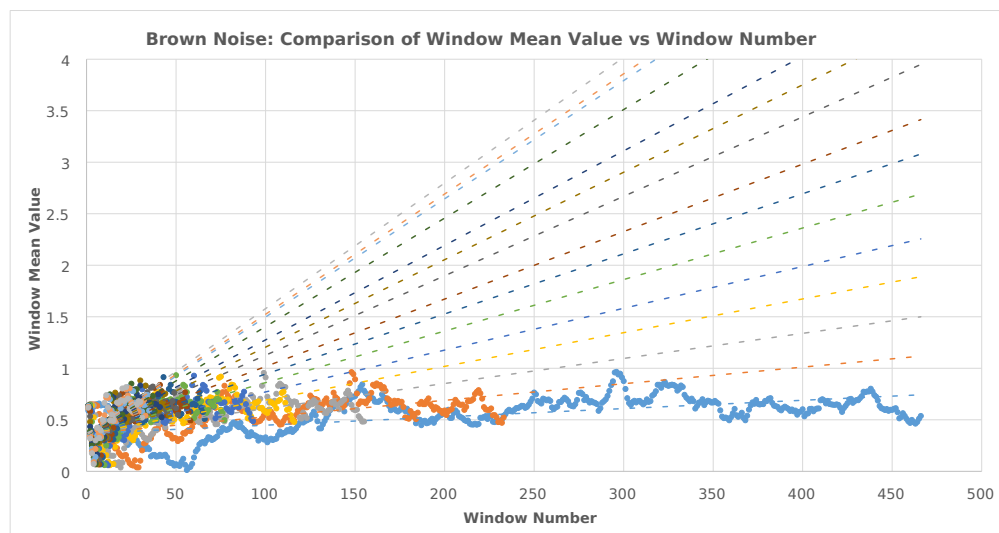
To illustrate the capability of this exploratory data analysis method we can visually compare the slopes of short time window means between White Noise and Brown Noise as shown in Fig. 5.15. The plot shows the heteroscedastic nature of the short time window means as the window size changes for Brown Noise by comparison to White Noise. The plot shown in Fig. 5.16 shows the means of short time windows vs window number for the received  $E_b/N_0$  from ExampleSat-165 to all the North American I-Gates. Notice the level of variance between the slopes as the window size changes.

Now that we have these slopes we can visualize the rate of change of slope with an increase in window size. We can see in Fig. 5.17 for White Noise that the mean value for slope is slightly heteroscedastic as the window size increases. The range of change of slope with short

time window size is small as indicated by the narrow change in scale from 0 to  $8 \times 10^{-5}$ . This demonstrates that the overall slope of the mean values for the short time windows sizes is small, which correlates with the notion that the change in mean is also small and thus the series may be stationary. We must keep in mind that the plotted points within Fig. 5.17, Fig. 5.18 and Fig. 5.19 are the slopes of the mean value of each short time window size across their respective time series.

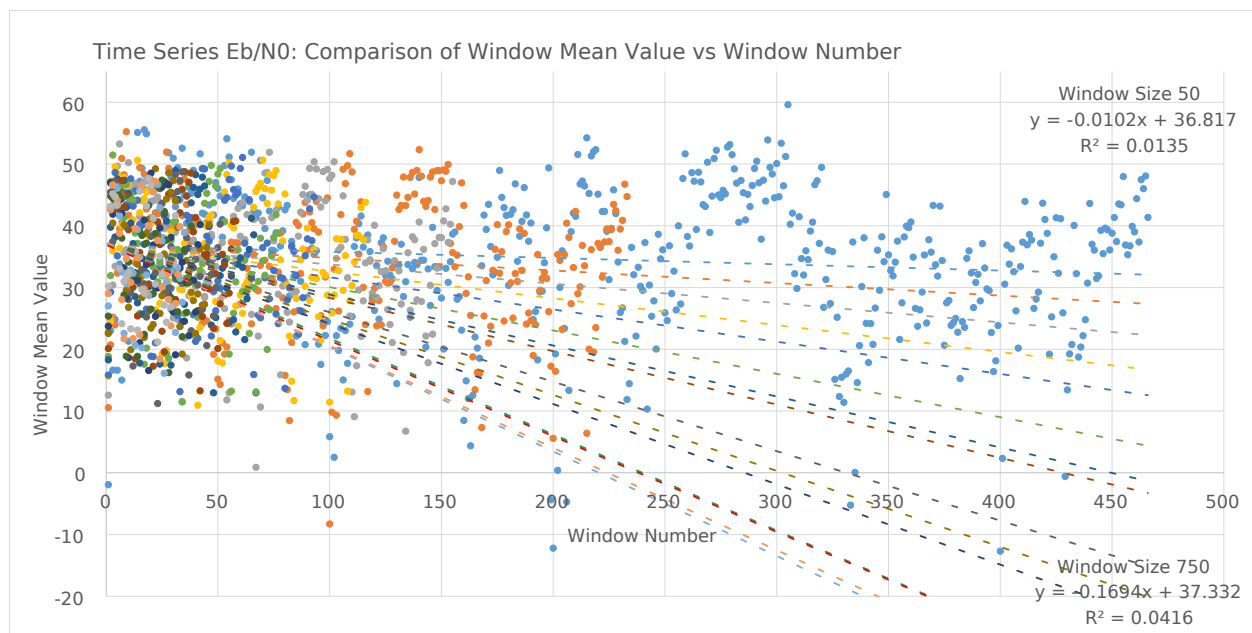


(a) White Noise.

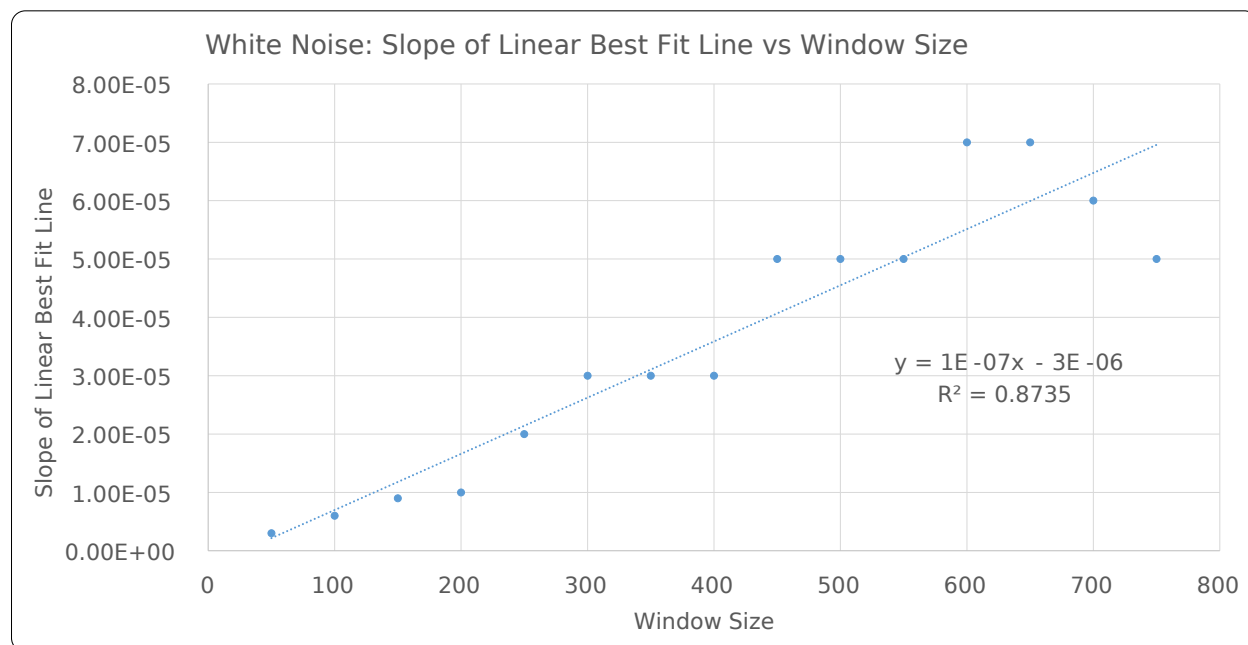


(b) Brown Noise.

**Fig. 5.15:** Plot of White and Brown Noise short time window means vs window number, axis adjusted for simple visual comparisons.



**Fig. 5.16:** Plot of  $E_b/N_0$  Window mean value vs window number with comparative slopes for all short time window sizes.



**Fig. 5.17:** Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.

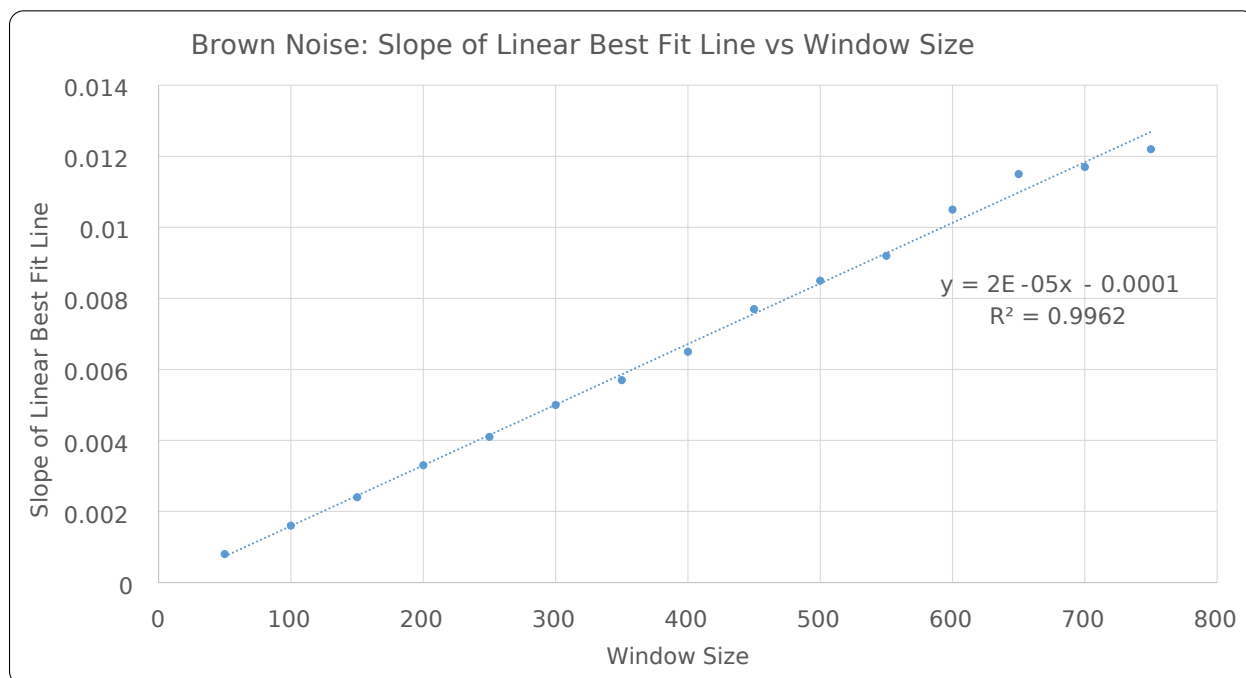
When we look at Fig. 5.18 we can see how the rate of change of slope for Brown Noise is roughly two orders of magnitude larger than that of White Noise. Interestingly the level of heteroscedastic tendency is lower but still apparent as the window size increases. It would be logical to imply that if we compare the slopes of short time windows of an unknown stationarity time series with those of White Noise or Brown Noise we may be able to classify the unknown series as either stationary or non-stationary.

Based on this premise we move onward to Fig. 5.19, noticing how the slope changes for received  $E_b/N_0$ . The negative direction of slope provides little additional insight other than the relative values of the underlying distribution of received  $E_b/N_0$ . Looking directly at the scale of each plot as a method of comparison is limited. Both the Brown Noise plot and the  $E_b/N_0$  plot share a similar scale for the the slope of the linear best fit lines rounded within two orders of magnitude ( $m \times 10^{-2}$ ), whereas White Noise requires five orders of magnitude ( $m \times 10^{-5}$ ). Thus we could attempt to say that to classify a known stationary series such as White Noise the slope of an estimated best fit line for the mean of sequential windows should fit within negative seven orders of magnitude ( $m \times 10^{-2}$ ). Such a small scale of magnitude indicates that the slope is small by comparison of a known non-stationary series as we would expect since the mean values should not show significant variance.

Within Fig. 5.19 we see that the rate of change of slope (the “slope of slopes” or second order derivative of the estimated means) is an order of magnitude larger than that of Brown Noise and three orders of magnitude larger than White Noise.

Based on what is shown we would be inclined to state that the unknown stationarity of received  $E_b/N_0$  appears non-stationary from the short time window non-stationary analysis as the series shares more characteristics in common with the known non-stationary series of Brown Noise. All three time series are visible in Fig. 5.20 where we can compare the slopes

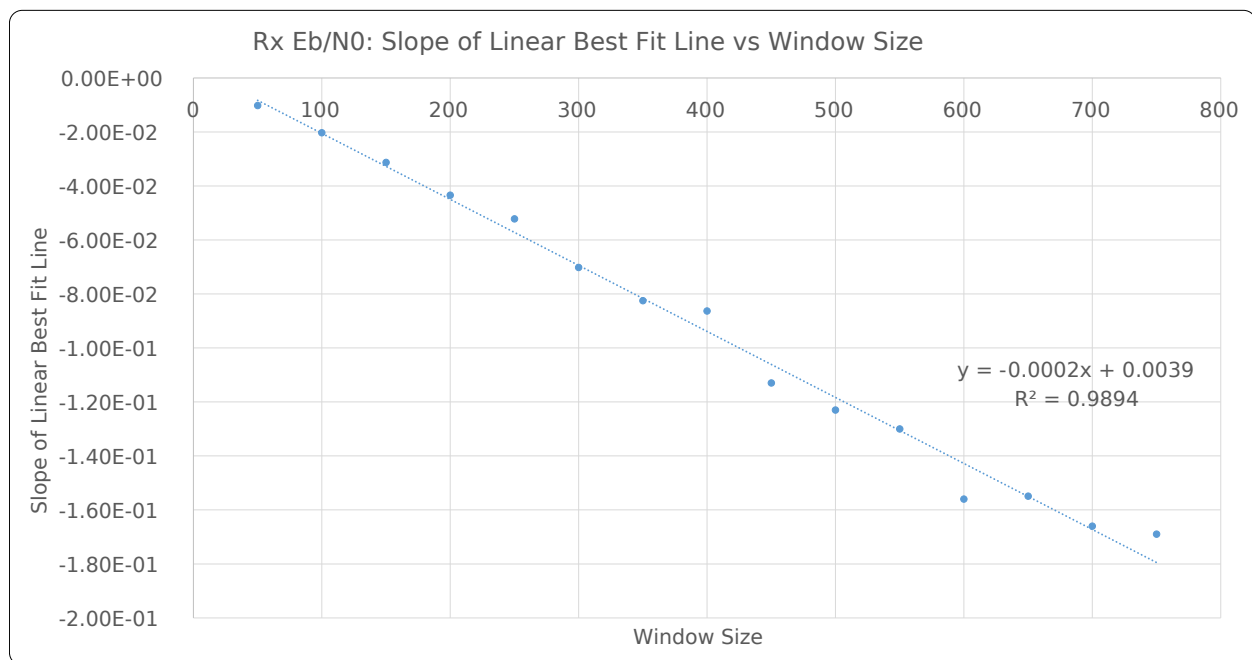




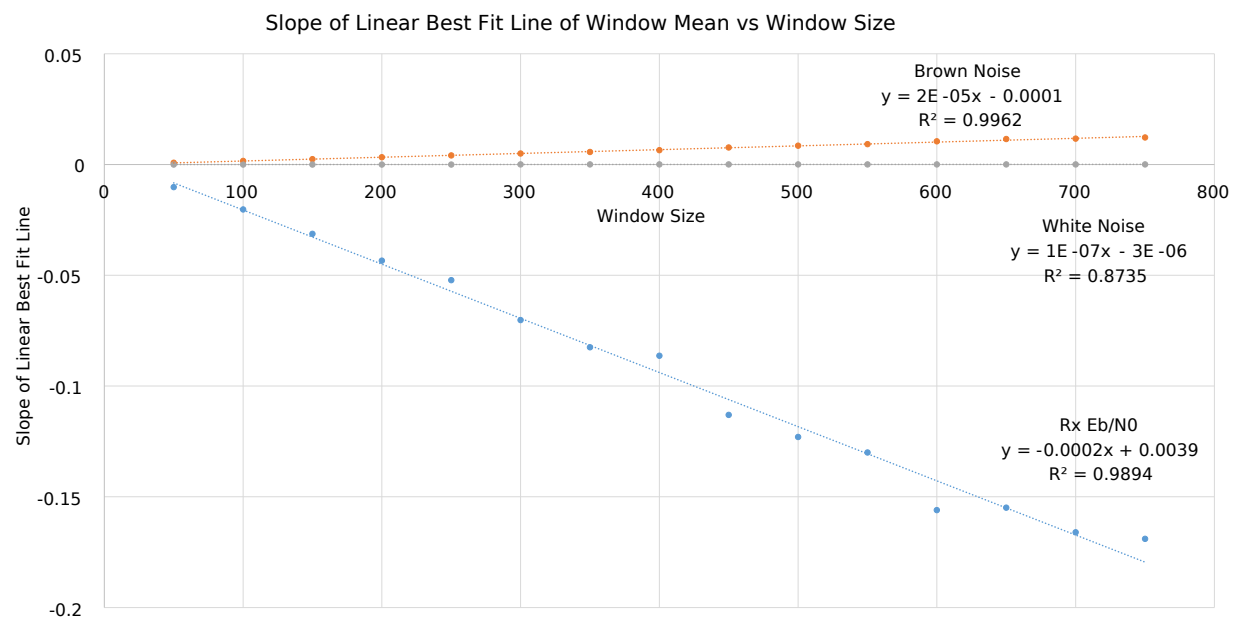
**Fig. 5.18:** Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.

of White Noise, Brown Noise and Rx  $E_b/N_0$ . At first glance this looks to be a promising technique for comparison between time series.

However, this method requires more rigorous investigation than graphical approximations. Does the short time window size have a statistically significant impact on the window mean? Can short time window analysis actually differentiate between known stationary White Noise and known non-stationary Brown Noise? This leads us to the use of Analysis of Variance (ANOVA).



**Fig. 5.19:** Plot showing the Slope of Linear Best Fit Lines with changes in Short Time Window Sizes.



**Fig. 5.20:** Plot showing all three time series Linear Best Fit lines for comparison.

### 5.4.2 ANOVA

Since we cannot use typical statistical tools to compare the sampled window data directly as we have concerns about each sampled window's stationarity and normality, we will need a level of abstraction for comparative purposes. What we are looking for is if the data series in question (White Noise vs Brown Noise vs Rx  $E_b/N_0$ ) is more significant than the size of the short time window itself. If the change in short time window size is statistically significant and the the origin/source of the data series is not, then we know that the technique of applying sampled window sizes maintains generality. This brings us to questioning the nature of the data series under test. If the method cannot distinguish between a known stationary series of White Noise, a known non-stationary series of Brown Noise and an unknown stationarity series of Rx  $E_b/N_0$ , all while the change in short time window size is statistically significant, then we must state that the method of Short Time Window Analysis of Non-Stationary Time Series is inconclusive at best.

**Table 5.5:** ANOVA: Single Factor: White Noise Window Size.

Groups	Count	Sum	Average	Variance
Window Size	15	6000	400	50000
White Noise Slope	15	0.000538	3.59E-05	5.31E-10

Source Of Variation	SS	df	MS	F	P-value	F crit
Between Groups	1200000	1	1200000	47.99999	1.57E-07	4.195972
Within Groups	700000	28	25000			
Total	1900000	29				

$H_0$ : Short Time Window Size has no significant impact on the slope of means at the 5% level.

$H_1$ : Short Time Window Size has a significant impact on the slope of means at the 5% level.

Based on the data presented in Table 5.5 we therefore reject the Null Hypothesis. Thus window size has a statistically significant impact on the slope of the short time window means of White Noise at the five percent (5%) level.

Based on the data presented in Table 5.6 we therefore reject the Null Hypothesis. Thus window size has a statistically significant impact on the slope of the short time window means of Brown Noise at the five percent (5%) level.

Based on the data presented in Table 5.7 we therefore we reject the Null Hypothesis. Thus window size has a statistically significant impact on the slope of the short time window means of the  $E_b/N_0$  time series at the five percent (5%) level.

Based on the data presented in Table 5.8 we fail to reject the  $H_0(rows)$  hypothesis. Therefore changing the data series (White Noise vs Brown Noise vs  $E_b/N_0$  time series) has no significant impact on the slope of the means of the Short Time Windows across time. Thus the different series under investigation using this method has no statistical impact beyond the five percent (5%) level.

Based on the data presented in Table 5.8 we can additionally reject the  $H_0(columns)$  hypothesis. Thus changing the Short Time Window Size has a statistically significant impact

**Table 5.6:** ANOVA: Single Factor: Brown Noise Window Size.

Groups	Count	Sum	Average	Variance
Window Size	15	6000	400	50000
Brown Noise Slope	15	0.1007	0.006713	1.46E-05

Source Of Variation	SS	df	MS	F	P-value	F crit
Between Groups	1200000	1	1199960	47.99839	1.57E-07	4.195972
Within Groups	700000	28	25000			
Total	1899960	29				

$H_0$ : Window Size has no significant impact on the slope of means at the 5% level.

$H_1$ : Window Size has a significant impact on the slope means at the 5% level.

**Table 5.7:** ANOVA: Single Factor:  $E_b/N_0$  time series data Window Size.

Groups	Count	Sum	Average	Variance
Window Size	15	6000	400	50000
White Noise Slope	15	-1.4083	0.006713	1.46E-05

Source Of Variation	SS	df	MS	F	P-value	F crit
Between Groups	1200563	1	1200536	48.02253	1.57E-07	4.195972
Within Groups	700000	28	25000			
Total	1899960	29				

$H_0$ : Window Size has no significant impact on the slope of means at the 5% level.

$H_1$ : Window Size has a significant impact on the slope of means at the 5% level.

on the slope of the means of the Short Time Windows beyond the five percent (5 %) level. This concurs with the single factor ANOVA results. From the above analysis we conclude that the size of the Short Time Window is statistically relevant to slope of the window means, rather than the actual data series under test. Short Time Window Analysis of Non-Stationary Time Series is predicated on the premise of finding a window size (or duration or area) such that the series appears to be stationary within said window. However, if the inherent stationarity of the underlying series has no significant impact on the sensitivity of the analysis it brings into question the overall utility of attempting to determine localized windows of stationarity through Short Time Window Analysis.

### 5.4.3 Normality Testing

When using ANOVA there are some key questions that need to be addressed for us to trust the results [BBN1996].

- Are we dealing with small sample sizes?
- Are we dealing with unbalanced sample sizes?
- Does the sample set contain outliers that create an apparent non-normality?

**Table 5.8:** ANOVA: Two Factor Without Replication: White Noise, Brown Noise &  $E_b/N_0$  time series data vs Window Size.

Window Size	Slope		
	White Noise	Brown Noise	$E_b/N_0$
50	3.00E-06	0.0008	-1.02E-02
100	6.00E-06	0.0016	-2.03E-02
...	...	...	...
750	5.00E-05	0.0122	-1.69E-01

Variation Source	SS	df	MS	F	P-value	F crit
Rows	0.012204	14	0.000872	0.805246	0.657059	2.063541
Columns	0.094932	2	0.047466	43.84652	2.37E-09	3.340386
Error	0.030311	28	0.001083			
total	0.137447	44				

$H_0(\text{rows})$ : Switching between data series has no significant impact on slope of means at the 5% level.

$H_1(\text{rows})$ : Switching between data series has a significant impact on slope of means at the 5% level.

$H_0(\text{columns})$ : Changing Short Time Window Size has no significant impact on slope of means at the 5% level.

$H_1(\text{columns})$ : Changing Short Time Window Size has a significant impact on slope of means at the 5% level.

- Are we dealing with unequal population variances?
- Is there a lack of independence between samples?
- Is there a lack of independence within the sample?
- Are we dealing with an entire sample set that is non-normal ?

Let us begin by looking at the sample sizes. Our actual test statistic of interest is the received  $E_b/N_0$  from ExampleSat-165 to the virtualized ground station network of APRS I-Gate stations over a 24 hour period. Our simulation in STK generated 23306 data points. Because this represents a very large sample size we may be able to use the Central Limit Theorem to our advantage. To avoid issues with unbalanced sample size we also generated the same number of points for both the White Noise and Brown Noise data series. Based on the plots shown in Fig. 5.5 and Fig. 5.12 we do not see any outliers that could pull our data series away from normality.

The F-test on which ANOVA is based is fairly robust against inequalities of variances when the sample sizes are equal, which further supports the need for both all of our data series being of equal size. We must acknowledge that the variance of the  $E_b/N_0$  time series data is larger than both of the generated noise series but its hoped that the large and equal sample sizes will mitigate any difficulties within inequalities of variance. Because we may be dealing with both inequal variances and non-normal data sets it would be typical to apply a transformation to the raw data before analysis. However, this could have an impact on the slope comparison method by dampening or amplifying the resulting slopes, and since we are working the a second order derivative (slope of slopes) this could compound such transformations. Furthermore, transformations of the raw data set would be specific to a

particular time series and thus the method would lose its generality which is the overarching goal of exploring this analytical method.

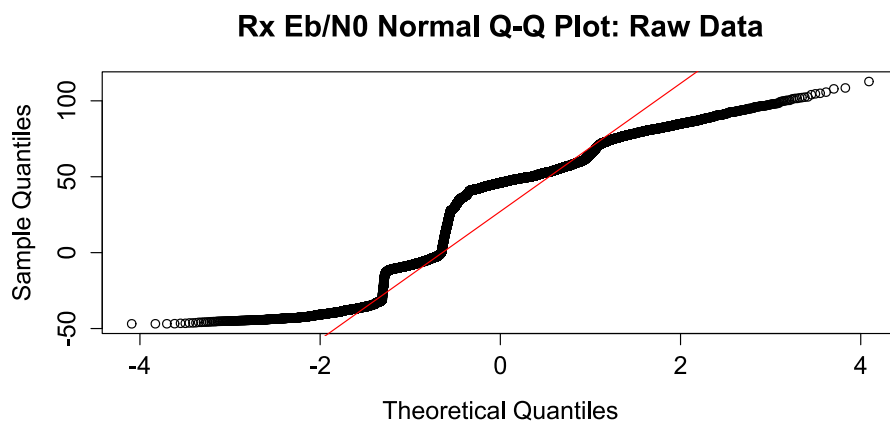
Next, let us address the issue of independence between samples. We know that for White Noise the data points are chosen independently by definition and are serially uncorrelated random values. This is not the case for Brown Noise as the data points within such a series are determined by random walk and thus are serially correlated. Subsequent points are randomly determined based on a seed value of the previous point. This implies that there is a lack of independence within the sample.

Because the Brown Noise series is serially correlated over time and our unknown stationarity series of Rx  $E_b/N_0$  may be as well, there is a concern that the data lacks independence between the samples. This is mitigated by the one-way blocked design for the experiment which also helps to minimize variances within the treatments and maximizes the potential for variance between treatments.

Finally we must investigate the normality of the data under scrutiny. Logically we know the data series of both White Noise and Brown Noise are not normally distributed because we used the TK95 function in R which uses power law to create each time series. This method was chosen for its simplicity of use and that it generates a constructively useful equidistant time series data set that we can compare to the equidistant simulation observables of  $E_b/N_0$  generated in STK. In particular, as we described in Sec. 4.3.1, the synthetic environment itself is a discrete event simulation in which we have arbitrarily chosen to increment our calculations at ten (10) second intervals, referred to as “Step Size” in STK, for the sake of computation duration. Additionally the use of a power law based function correlates directly to the generated data series’s spectral density which is of interest for further stochastic study.

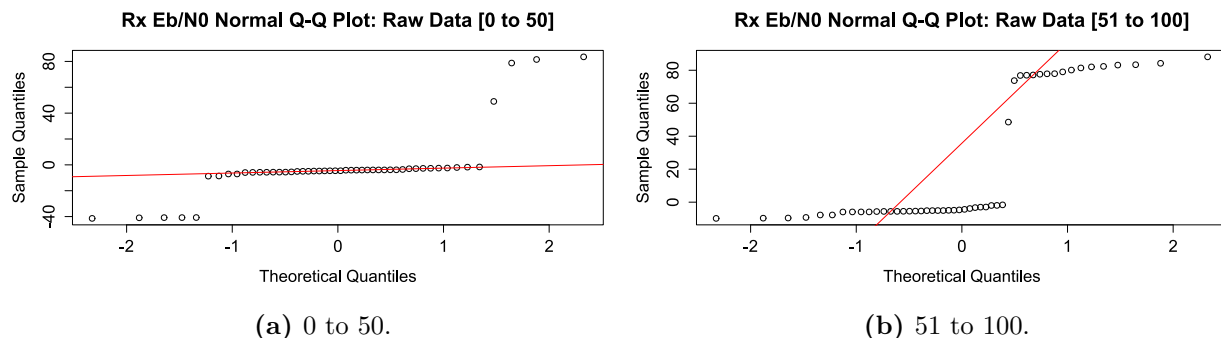
### 5.4.3.1 Normal Q-Q Plots

Investigating normality of a data series is often initially performed by graphical interpretation of a Normal Q-Q plot as shown in Fig. 5.21. Quantile-Quantile probability plots are a graphical method for comparing theoretical idealized quantile data against the quantiles of our sample data from any of our selected time series data sets.



**Fig. 5.21:** Normal QQ Plot of Rx  $E_b/N_0$  raw data series.

This plot can tell us generalities of the Rx  $E_b/N_0$  such as how it does not follow a the best fit normal distribution line. The outer quantiles are skewed in the opposite directions from each other while the median is positively skewed. We can see that the data has several underlying gaps shown by how it traverses the line of normality. This is expected based on how ExampleSat-165's orbital mechanics alters the received  $E_b/N_0$  during a 24 hour period. But we need to understand the normality of the time series data with respect to window size.



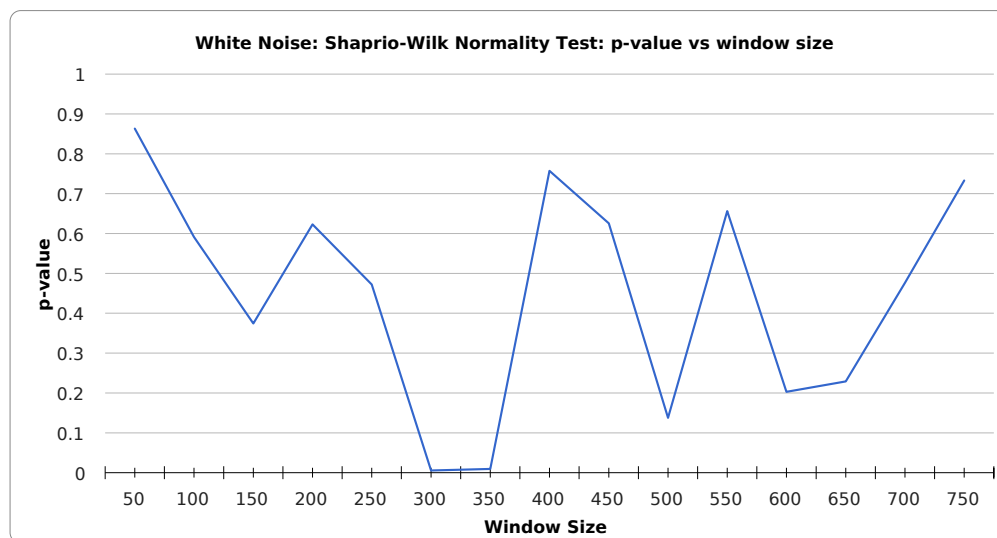
**Fig. 5.22:** Normal Q-Q Plots of Rx  $E_b/N_0$  from 0 to 50 and 51 to 100.

From Fig. 5.22 we see a comparison of normality for window sizes for the Rx  $E_b/N_0$  time series of the first 100 data points. It's clear that Fig. 5.22b is bimodal compared to Fig. 5.22a. Interpreting Q-Q plots is non-trivial. It is also clear that manually evaluating normality for all windows, at all of the window sizes across three different time series of data sets is impractical. This leads us to the Shapiro-Wilk normality test [Shapiro1965].

### 5.4.3.2 Shapiro-Wilk Normality test

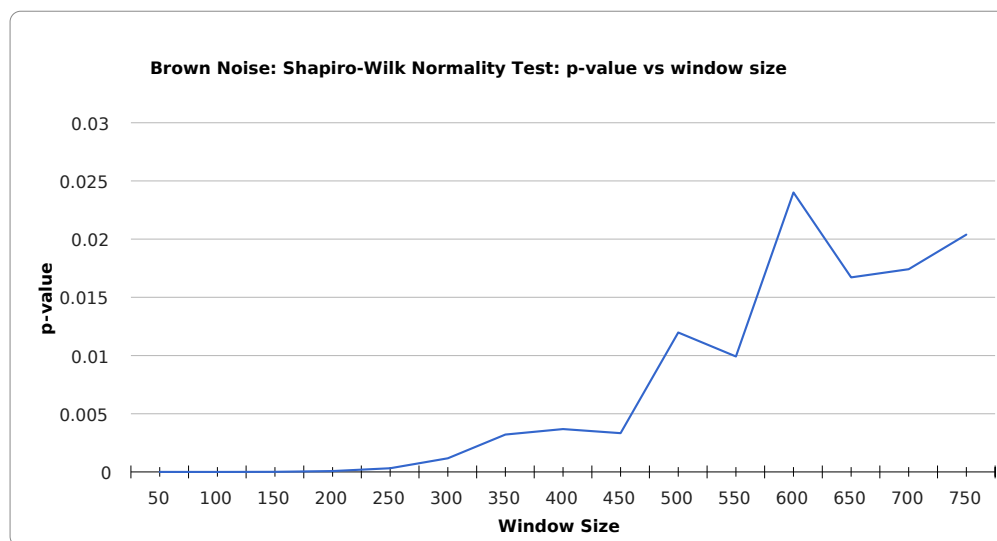
In 1965 Samuel Sanford Shapiro and Martin Wilk published “*An analysis of variance test for normality (complete samples)*” in the journal *Biometrika*. Their frequentist statistics approach to testing normality is based on the null hypothesis that a sample came from a normally distributed population. If the determined p-value from the test is less than the chosen alpha level (by convention this is typically a significance level of 0.05) then we reject the null hypothesis and thus must conclude that the data is not normally distributed. This provides us with a numerical benchmark such that we can calculate a mean the p-value for each short time window size and compare across window sizes. The added advantage is that the Shapiro-Wilk test is a simple function call within the R programming language. This test is popular as evidenced by approximately 81300 results when searching for “Shapiro-Wilk normality test” via Google Scholar as of 27 Sept. 2016. The tabulated results are available in Appendix D.

Looking at Fig. 5.23 we see that most of the p-values across the set of window sizes are above the alpha level of 0.05. Only the window sizes of 300 and 350 had low enough p-values (0.005517 and 0.009576 respectively) would lead us to reject the null hypothesis that the data was not normally distributed at those sizes. The remainder of the p-values were well above the alpha of 0.05 and thus fail to reject the null hypothesis that the series is



**Fig. 5.23:** White Noise Shapiro-Wilk Normality Test p-value vs Window Size.

normally distributed. This lead us to state that at those short time window sizes the series could be considered normal. This corresponds well since we know that the White Noise series is random and serially uncorrelated. We also see that the there is no apparent pattern to the change in normality as we change the window size which is a result of the serially uncorrelated random values mentioned earlier.

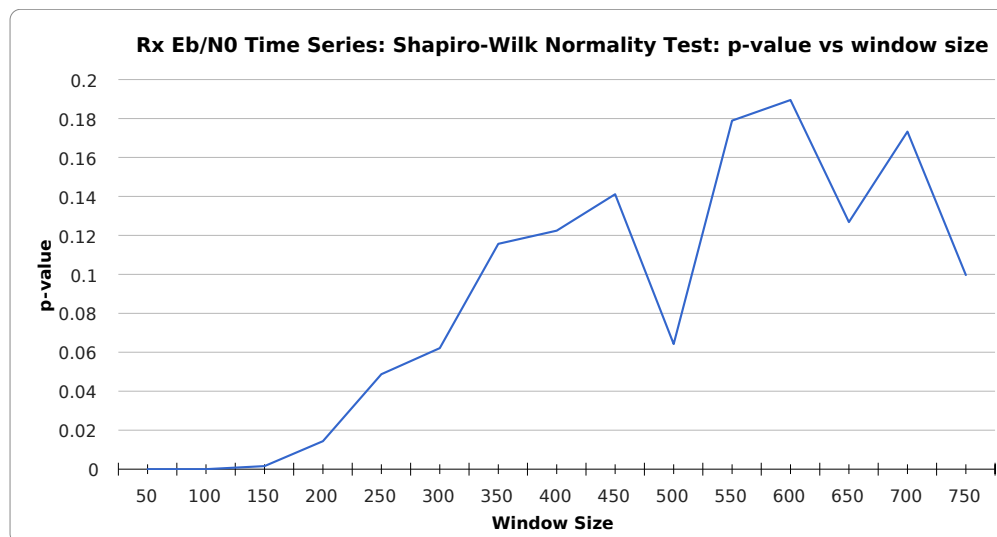


**Fig. 5.24:** Brown Noise Shapiro-Wilk Normality Test p-value vs Window Size.

Moving on to a plot of Shapiro-Wilk p-values with short time window size for Brown Noise in Table D.4 we find that all of the points within it are below the 0.05 alpha level. Thus we must reject the null hypothesis that the Brown Noise series is normally distributed. We can also see that normality is not achieved regardless of how much aggregation occurs as we increase the short time window size. Since we have an intuitive understanding that across the entire series Brown Noise is not normally distributed, unlike White Noise, we can now



state that the Shapiro-Wilk normality test is capable of determining normality for an known stationary series and can identify non-normality for a non-stationary series.



**Fig. 5.25:**  $E_b/N_0$  Time Series Shapiro-Wilk Normality Test p-value vs Window Size.

Finally we can examine the Rx  $E_b/N_0$  time series data in Fig.5.25. We see that there is a point when the short time window reaches 250 that the p-values remain above the critical decision alpha level of 0.05. Thus we can say that if the short time window is  $\geq 250$  samples then the Rx  $E_b/N_0$  series can be considered normally distributed. This has an impact on the available statistical methods used to evaluate, measure, and transform the Rx  $E_b/N_0$  time series by setting the lower boundary of the short time window size for analysis of the received signal.

Note that the Shapiro-Wilk test does not determine stationarity itself, merely if the series or short time window in question is normally distributed. To determine the stationarity of a series of a short time window we will use the Phillips-Perron Stationarity Test outlined in Sec. 5.5.

## 5.5 Phillips-Perron Stationarity Testing

The Dickey-Fuller (DF) test proposed by David Dickey and Wayne Fuller in their 1979 paper “Distribution of the estimators for autoregressive time series with a unit root” tests the null hypothesis that there is a unit root present in an autoregressive model [Dickey1979]. If the series is stationary or trend stationary then it has a tendency to return to a constant mean. Based on this logic the DF test suggests that large values will tend to be followed by smaller values and vice versa. An Augmented version of the DF test (ADF) with allows for higher orders of lag within the autoregressive process. The ADF is based on the premise that if the time series is integrated then the lagged level of the series will not provide relevant information in predicting changes in the variate other than the one obtained from the lagged change. As such the ADF requires that the lag length  $p$  must be determined/assumed prior to applying the test.

As discussed in Sec. 2.6.2, the Phillips-Perron Stationarity test was originally proposed by Peter C.B. Phillips and Pierre Perron in their 1988 paper “Testing for a unit root in time series regression” in the journal *Biometrika* [Phillips1988]. Their paper outlined a null-hypothesis in which a unit root is present within the data series. Unit roots are features of a process that are emergent features over time. They proposed that if one (1) is a root of the process’s characteristic equation then the process is non-stationary. In effect, can the time series be integrated of order one (1)? The Phillips-Perron (PP test) is non-parametric which enables us to not make as many assumptions about the data itself such as the lag order as required by the ADF test.

In his 1998 book “Elements of Forecasting” Francis X. Diebold with the University of Pennsylvania states “The basic idea of Phillips-Perron tests is to estimate a Dickey-Fuller regression without augmentation, and then to correct the Dickey-Fuller statistics for general forms of serial correlation and/or heteroscedasticity that might be present...” The PP test has been shown to work well with unspecified autocorrelation and heteroscedasticity. The PP test can be viewed as an extended form of Dickey-Fuller statistics that has been made robust against serial correlation by using the Newey-West heteroscedasticity and an autocorrelation-consistent covariance matrix within its estimator. A detailed description or derivation of the test is beyond the scope of this thesis. We can focus on the impact of Phillips and Perron’s their work as our justification for exploring the use of their test. Their 1988 paper has been cited 12884 times as of November 10, 2016. Much of this work is within fields related to econometrics, which often focuses its analysis on the predictability of time series data such as stock prices or energy usage. For examples of PP test usage outside of economics we can look at least two cases. In their 2013 paper “Blood Glucose Level Prediction Using Physiological Models and Support Vector Regression” [Bunescu2013] Bunescu *et al.* used the PP test within the R statistical function *auto.arima* “which uses the Bayes information criterion to determine the orders  $p$  and  $q$  of the autoregressive and moving average components, and the Phillips-Perron unit root test for determining the order  $d$  of the difference component.” In her 2014 doctoral dissertation Parastoo Qarabaqui, with Northeastern University, tested for stationarity of her derived data sets related to underwater acoustic communications channels [Qarabaqi2014].

Since the PP test is based upon a DF test it generates Dickey-Fuller  $z$  value (based on the chosen level of significance,  $\alpha$  which is typically 0.05), and a  $p$ -value. If the Dickey-Fuller  $z$  statistic is large, as an absolute value, while the  $p$ -value is less than our level of significance  $\alpha$  then we can infer that there is strong evidence to reject the null hypothesis that there is a unit root to the data series.

From here we must see how the PP test works with our known stationary White Noise series and our known non-stationary Brown Noise series. The PP test is provided by a the function *pp.test* within the *tseries* package [Trapletti2016] within R. The settings for the test are left at their default values so that the lag truncation calculation remains  $\text{trunc}(12 \cdot (n/100)^{0.25})$ . Since the size of each time series White Noise, Brown Noise and Rx  $E_b/N_0$  is the same the truncation lag parameter will be calculated at 15 for each case.

Applying the *pp.test* function to our White Noise series as shown in Fig. 5.12a the results of the Phillips-Perron Unit Root Test are:

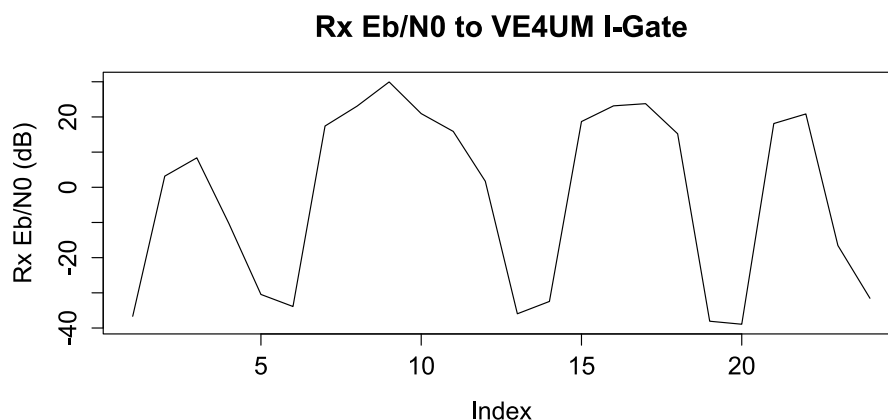
White Noise

- Dickey-Fuller  $Z(\alpha) = -23320$
- Truncation lag parameter = 15
- p-value = 0.01
- alternative hypothesis: stationary
- Warning message: In `pp.test(white)` : p-value smaller than printed p-value

Based on the large Dickey-Fuller  $Z(\alpha)$  absolute value of  $|-23320|$  and the p-value  $< 0.05$  we have enough evidence to reject the null-hypothesis and therefor take the alternative hypothesis to conclude that the White Noise series is stationary. This result aligns well with our knowledge that by definition the mean and autocovariance of randomly determined White Noise is stationary. The warning message states that the computed statistic is outside the table of critical values (Tables 4.1 and 4.2 on page 103 of Banerjee *et al.*) used as reference [Banerjee1993] for the calculations. The message implies that the p-value is too small for further interpolation which occurs for any calculated p-value of this size.

Predictably the PP test provides a much smaller Dickey Fuller  $Z(\alpha)$  value for the Brown Noise series shown in Fig. 5.12b while the p-value is above the 0.05 threshold. This implies the we fail to reject the null hypothesis that there is a unit root present. Thus we conclude that the Brown Noise series is non-stationary.

The capabilities of the PP test were further explored by looking at an obviously non-stationarity series such as the received  $E_b/N_0$  from ExampleSat-165 to a single I-Gate such as VE4UM as shown in Fig. 5.26 followed by the full Rx  $E_b/N_0$  time series from ExampleSat-165 to all 842 of the virtualized ground station network I-Gate nodes as shown in Fig. 5.5.



**Fig. 5.26:** Non-stationarity of the received  $E_b/N_0$  for ExampleSat-165 to the VE4UM I-Gate.

The results of the PP tests are summarized in Table 5.9. This table demonstrates that the PP test can differentiate between known stationary and known non-stationary series. Furthermore it has classified the full Rx  $E_b/N_0$  time series as stationary while a portion

**Table 5.9:** Phillips-Perron Stationarity Testing.

Series	Figure	Dickey-Fuller Z(alpha)	Lag parameter	p-value	Stationary
White Noise	Fig. 5.12a	-23320	15	0.01	yes
Brown Noise	Fig. 5.12b	-14.929	15	0.2789	no
Rx $E_b/N_0$ single I-Gate	Fig. 5.26	-14.803	2	0.1534	no
Rx $E_b/N_0$ Full Network	Fig. 5.5	-31561	15	0.01	yes

of the represented series from ExampleSat-165 to the single VE4UM I-Gate station was classified as non-stationary.

The PP test has demonstrated that by the received  $E_b/N_0$  from ExampleSat-165 to a single ground station node is non-stationary, but that by creating a virtualized ground station network the total received  $E_b/N_0$  from ExampleSat-165 to the entire set of nodes can be considered stationary. Based on this premise it would be possible to conduct more conventional signals analysis techniques such as Fourier transforms, which previously would not be a valid form of analysis as its dependent on the assumption of the underlying system is time invariant. Our next task is to determine if the PP test remains useful when we look at short time windows of the same time series data sets.

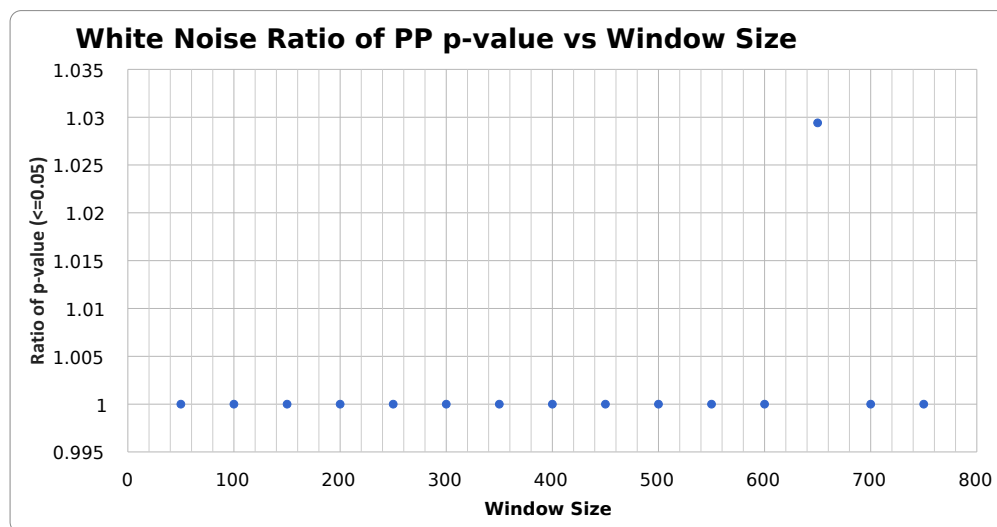
### 5.5.1 Phillips-Perron p-value ratios

To establish if the PP test still has utility when looking at short time windows of our non-stationary time series we need to see if the stationarity of a series changes based on the size of the window. Conceptually we will slice each time series into equal incremented window sizes as we did with the slope comparison method in Sec. 5.4.1. We will then use the PP test on each short time window and count the number that are considered stationary based on the p-value of the test. For comparative purposes we will look at the ratio of stationary to non-stationary short time windows based on their size. We define the Ratio of PP p-values as the ratio of p-values  $\leq 0.05$  vs the total number of possible windows of that short time window size.

By example, if our time series of 23308 points is sampled at a short time window size of 50 samples this results in 466 whole windows (rounded to the nearest whole window). Of this set, if all 466 windows were PP tested and every one resulted in a p-values of  $\leq 0.05$  this provides a ratio of 1 for that window size. This Ratio of PP p-values tells us a relative portion of the short time windows of a given size that are considered stationary. As the ratio approaches one (1) we can consider that the short time window size in question is stationary for the entire series.

The plot in Fig. 5.27 shows that for White Noise the PP test rejects the null hypothesis that a unit root is present in the data for most of the short time window sizes and therefore the series is stationary at each of these window sizes. We see spike in the ratio value at a window size of 650. We see similar results at 550 for Brown Noise and the final Rx  $E_b/N_0$  time series data at 650 as well. This represents an aliasing error for the data based on the sampled window size.

The plot in Fig. 5.28 shows that for Brown Noise the PP test fails to reject the null



**Fig. 5.27:** White Noise Phillips-Perron Stationarity Test p-value vs Window Size.

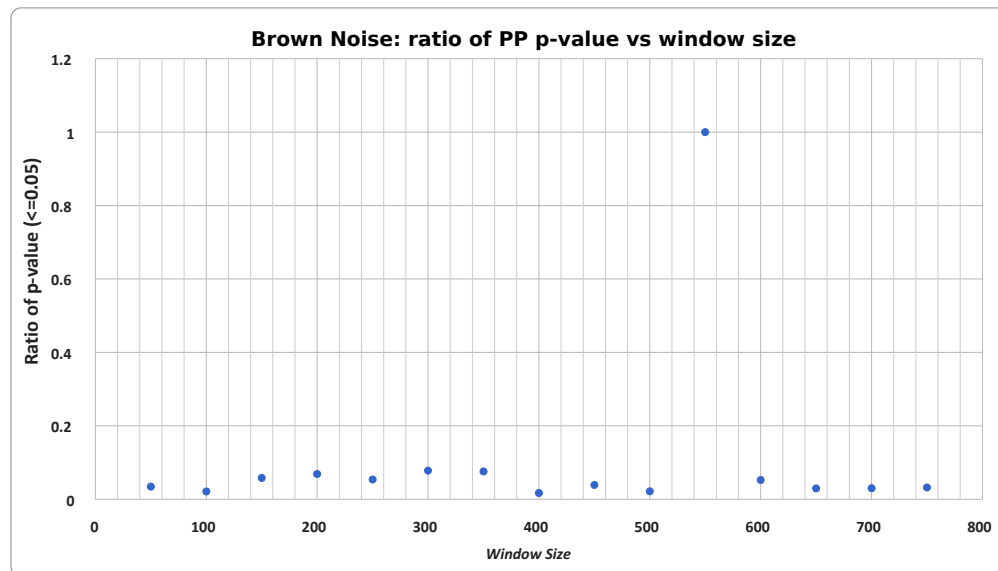
hypothesis that a unit root is present in the data at most of the short time window sizes. Therefore the series is consistently non-stationary (aside from the aliasing error at 550).

The plot in Fig. 5.29 shows that for the Rx  $E_b/N_0$  time series data the PP test rejects the null hypothesis that a unit root is present in the data and therefore the series is mostly stationary (aside from the aliasing error at 650). Of interest is the dip in the ratio at 50. This indicates that a smaller set of window sizes from 1 to 100 may be required to fully explore the stationarity of the data set.

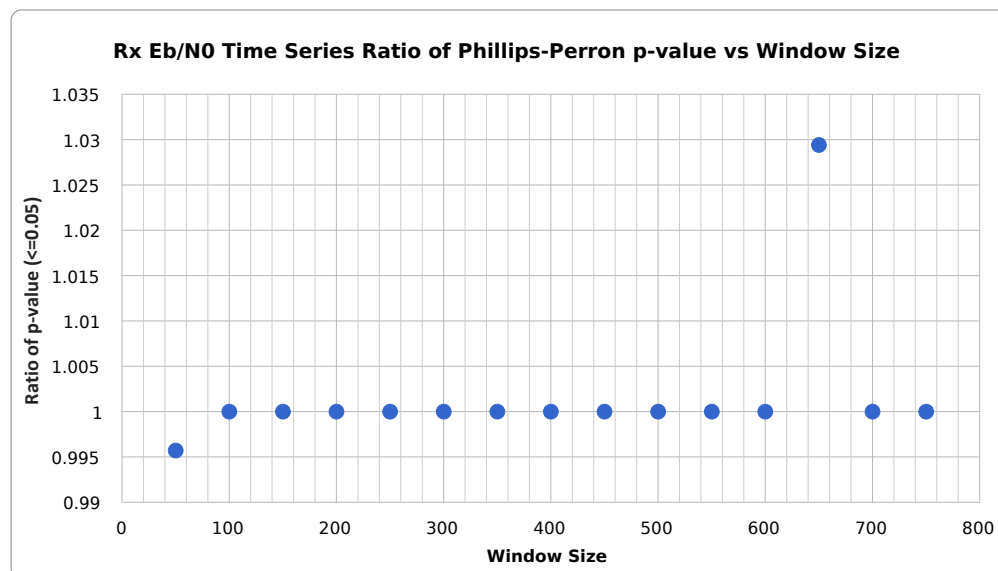
For comparison purposes all three data series are shown in Fig. 5.30. This graph clearly shows how similar White Noise and Rx  $E_b/N_0$  are across all window sizes. By comparison, our known nonstationary Brown Noise does not appear to have any window in which it appears stationary. Note that since the issue of aliasing is still present within the data set due to the nature of sampling the time series.

What the results and plots in Sec. 5.5 has shown us is that the Phillips-Perron test has a high level of utility for classifying signals between stationary or non-stationary, whether analyzing the entire time series or short time windows of a series. We can also state that by creating a virtualized ground station network of the North American APRS I-Gate nodes that the total received signal can be considered stationary when compared to the received signal for a single I-Gate node.

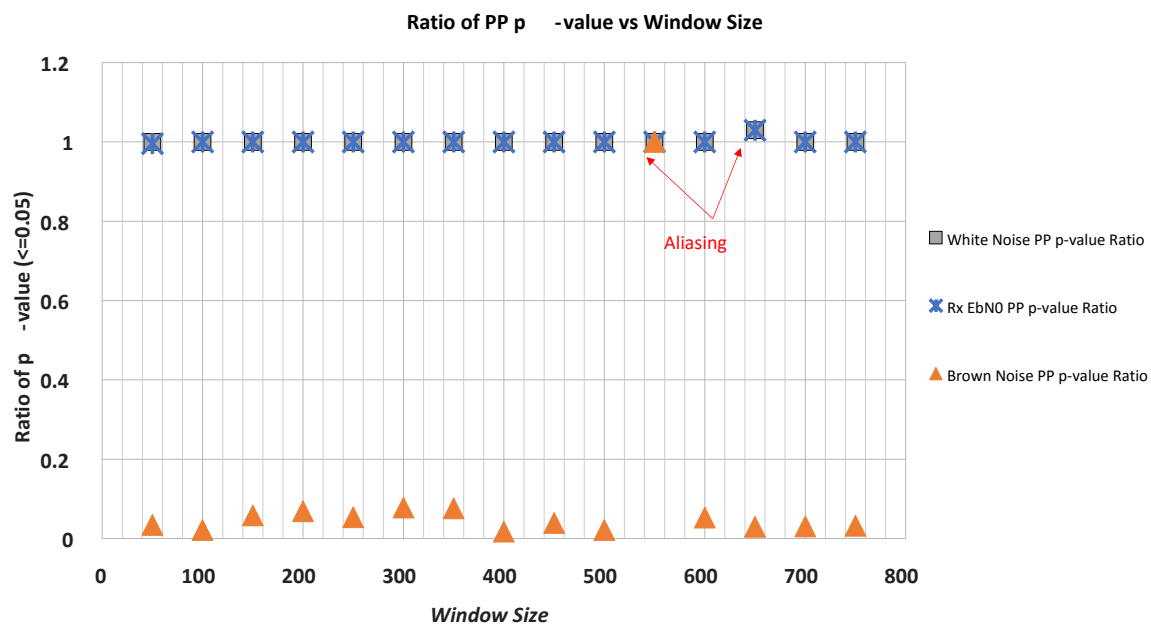
In effect the virtual ground station network overcomes two fundamental problems if not using single ground station Yagi-Uda tracking antenna system as illustrated in Fig. 2.7 or Fig. 4.7. The nonstationarity of the received  $E_b/N_0$  due to slant range is mitigated by having additional I-Gates receiving the same signal from different ranges at the same time. Furthermore the added I-Gate nodes overcome issues associated the using antennas of differing polarization. With 842 ground stations, at some point during the orbital period, there will be more than one node with an antenna orientation that allows reception with ExampleSat despite no prior arrangement of polarization.



**Fig. 5.28:** Brown Noise Phillips-Perron Stationarity Test p-value vs Window Size.



**Fig. 5.29:** Received  $E_b/N_0$  time series Phillips-Perron Stationarity Test p-value vs Window Size.

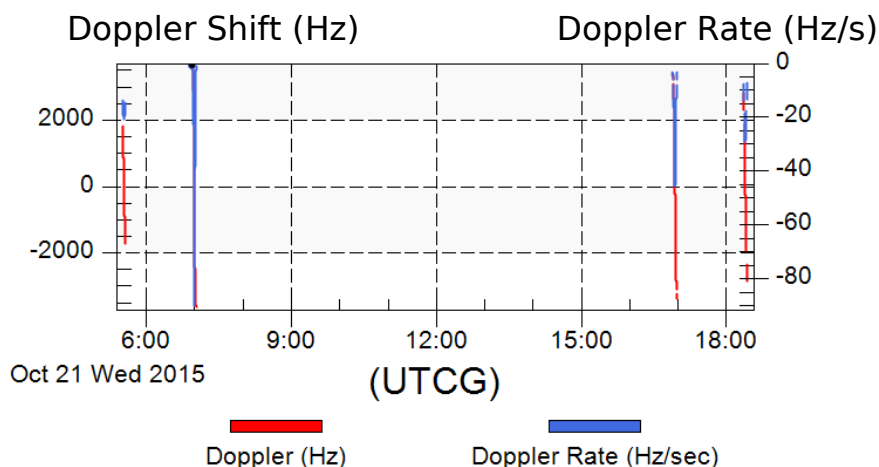


**Fig. 5.30:** Ratio of Phillips-Perron Stationarity Test p-value vs Window Size for all three time series.

## 5.6 Doppler Shift Results

We began Sec. 2.7 discussing the nature of Doppler shift. In particular Sec. 2.7.1 explored the impact of induced errors as the mark (binary 1) and space (binary 0) notch filters on the I-Gate stations do not adjust for Doppler shift. A planned experiment to emulate and study this problem was described in Sec. 3.4. Although the planned experiment was not completed the key question remains; What level of Doppler shift and/or Doppler rate is tolerable for the typical APRS I-Gate node to correctly interpret packets. To begin with such an experiment the boundaries of the problem space need to be explored, which leads us back to our simulation.

Within the STK simulation its possible to calculate not only the instantaneous Doppler Shift ( $Hz$ ) as seen by a node, but also the rate of change of Doppler Shift called the Doppler Rate ( $Hz/s$ ). These values are available by creating customized reports and charts as discussed in Sec. 5.2. The plot shown in Fig. 5.31 provides both values as perceived by a the single I-Gate node VE4UM over a 24 hour period starting at 05:00 Oct 21 2015. The red line uses the left hand axis for the perceived Doppler shift, whereas the blue line uses the right axis for the reprieved Doppler rate. Notice that the plot includes not only the total required bandwidth for the Doppler shifts but also the time at which they will occur.



**Fig. 5.31:** Doppler Shift and Doppler Rate seen by VE4UM's single I-Gate from ExampleSat-165.

Within this scenario the VE4UM I-Gate station experienced a maximum Doppler rate of  $-90.563655 Hz/s$ , a minimum Doppler rate of  $-0.559804 Hz/s$  resulting in a mean of  $-19.510081$ . This provides us with the boundary regions for experimentation with hardware emulation using software defined radios.

If we look at VE4UM as a single I-Gate ground station downlinking from ExampleSat-165, VE4UM would experience the following Doppler Shift to the RF carrier on 144.39 MHz.

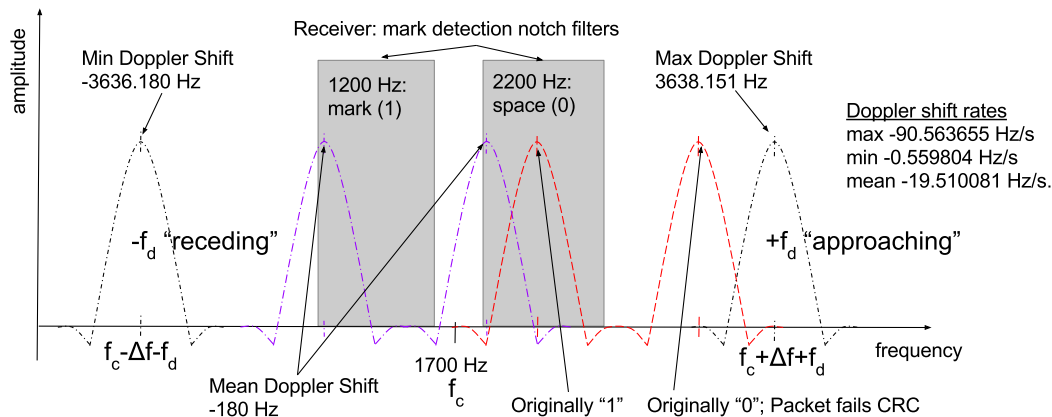
The left column of statistics in Table 5.10 represent the *instantaneous apparent frequency* of the received signal according the VE4UM I-Gate station. The right column shows the *rate of change of the apparent frequency* of the received signal according to the VE4UM I-Gate station. With a one to one ratio of applied Doppler shift from the RF carrier band to the



**Table 5.10:** Doppler shift and Doppler Rate received by VE4UM I-Gate from ExampleSat-165 over 24 hours.

	Doppler Shift (Hz)	Doppler Rate Shift (Hz/s)
Minimum	-3636.180	-0.559804
Maximum	3638.151	-90.563655
Mean	-180.173	-19.510081

audio base band demodulated signal we can see the apparent issues induced within our data stream due to ExampleSat's orbital motion.

**Fig. 5.32:** The impact of Doppler shift on the demodulated AX.25 audio baseband signal.

Revisiting our discussion of AX.25 in Sec. 2.4.3, specifically Fig. 2.35 we can edit the figure based on the values from Table 5.10 to create Fig. 5.32 which shows the impact of Doppler shift on the demodulated audio baseband signal. Clearly the minimum and maximum instantaneous Doppler shift could push the received signal outside the notch filter for a "Zero" or a "One". The base band audio notch filter for each node is dependent on the hardware used by each node. In the heterogeneous APRS I-Gate network we would expect a distribution of widths of notch filters. Note that the mean Doppler Shift of -180 Hz may still be within the notch filter for the demodulated base band signal to be interpreted correctly overall.

Let us look at a worst case scenario of maximum Doppler shift rate of -90.563655 Hz/s across a one second stream of 1200 bps data. This becomes 13.250 Hz/s for each bit within the data stream across a one second interval. The Doppler shift rate represents the rate at which the RF carrier deviates from the centre frequency of 144.39 MHz as time progresses. Applying this Doppler shift rate from the RF carrier band to the audio base band signal at a one to one ratio we begin to appreciate its limited impact on the demodulation of individual bits within the data stream.

From Table 2.3, we see that the total packet size is 332 bytes or 2656 bits. Which takes 2.2133 seconds to transmit at 1200 bps. This equates to -200.4475 Hz of total frequency deviation across the full duration of the packet based on our -90.563655 Hz/s maximum Doppler shift rate. Given that there is 1000 Hz of separation between "Zero" and "One" for the AX.25 base band bits shown in the figure above, it is reasonable to assume that a transmission of a packet that began without apparent Doppler shift would be received without error.

We should also keep in mind that the older Bell 202 standard, on which AX.25 is based, was developed for twisted pair telephony during a period in which more inherent noise was expected than in current telecommunications systems. Thus we would assume that the audio base band notch filters would be "comparatively" wide. This design principle would then carry over to the current generation of I-Gate node hardware in order to maintain compatibility with older equipment.

Again this is where the idea of the virtualized ground station network shows its advantage. We know that Doppler shift is due to changes in slant range as the satellite approaches or recedes from the I-Gate node. We also know that the received  $E_b/N_0$  is proportional to the change in slant range, which was mitigated by the additional nodes within the virtualized ground station network. Thus the changes from induced Doppler shift would be mitigated by similar proportions. Not all nodes would hear the same apparent Doppler shift, some of them would receive packets with limited Doppler shift and thus be without error.

Based on the maximum Doppler shift, we can see that its possible for some nodes to receive Doppler shifts of approximately 1000 Hz. This example would place a value of "One" within the notch filter of a "Zero". As we see in the figure above, such an extreme situation would also pull the corresponding "Zero" outside of the other notch filter, causing the packet to fail its CRC and thus be discarded. This works to the virtualized ground station network's advantage as it discards packets with errors due to extreme Doppler shift. Since APRS is a best effort style protocol it does not request retransmissions and thus limits flooding of the network due to packet errors. Again this works out to ExampleSat-165's favor as the error created by its Doppler induced frequency shift are not causing ground stations or I-Gates to call for repeated transmissions due to packet CRC failures.

Combining together the effects of Doppler shift, Doppler shift rate, and the band gap between notch filters, we can now see that its mostly likely packets will either be interpreted correctly or fail the CRC. This is a key property of the Flag field of the AX.25 packet structure. During the 2.2133 seconds it takes to transmit one packet, is there enough induced Doppler shift at the given Doppler Shift rate to cause the packet to fail the CRC because the Flag field was received incorrectly? Clearly many I-Gate nodes will received packets which fail the CRC. But a number will not have the same induced Doppler shift and thus not fail the CRC. Additionally, block errors which could pass through a CRC are more likely to fail within this situation due to Doppler shift.

At first glance, Doppler induced errors appears to be a problem which solves itself within the virtualized ground station network. However, much of the preceding is based on the width of the notch filters for a heterogeneous network which has not been measured. This was the initial reasoning behind the Software Defined Radio experiment described in Sec. 3.4.

## 5.7 Summary

In Chapter 4 we explored how a combination of community surveys, far field antenna models, orbital elements, transmitters, receivers and captured APRS packets combined together to form a discrete event simulation. In Chapter 5 we can see how everything in the preceding chapters has led us to useful data with which we characterized the capabilities of the virtualized ground station network. By looking at the received  $E_b/N_0$  and Eq. 5.1 we can then predict how well packets or entire science payload data sets will be received (Eq. 5.3, Eq. 5.4, Eq. 5.5 and Table 5.4 ).

Diving deeper into the Rx  $E_b/N_0$  time series from a statistical analysis standpoint required us to compare it with a known stationary series of White Noise, and a known non-stationary series of Brown Noise which were generated in the R statistical programming language. We examined the stationarity of each time series via short time windows, first through comparisons of slope, then by Phillips-Perron test. The method of Slope comparison proved to be problematic in that it could not differentiate between known stationarity and known non-stationary time series. This difficulty was further compounded by drifts in normality as the short time window size increased. But we should note that normality did improve with the increase in window size. From this analysis we can also conclude that the minimum short time window size for analysis of Rx  $E_b/N_0$  should be  $\geq 250$ .

The PP tests were shown as capable of correctly classifying known stationarity and known non-stationary time series across changes in short time window sizes. Furthermore the Rx  $E_b/N_0$  time series was shown to be stationary as an entire series as well as for the majority of short time window sizes.



## 6 Thesis Conclusions

At first this thesis attempted to answer a basic functionality question related to using APRS and CubeSats. “Why do not we use a normal I-Gate to download data from a CubeSat?” From this simple starting point many more questions needed answers. Is such a link even possible? What would be the advantage compared to a normal ground station with a large directional antenna? Would that flood the network with packets that bounce back and forth across the RF link? How would you model the satellite link? Would the aggregated network design provide a capability that was not possible before? How accurate can this be modeled? How can this concept be extended?

To answer the initial question, this thesis shows that, its possible for you to download data from a CubeSat via an ordinary APRS I-Gate. This does imply that you do not have to build your own complicated ground station. It further implies that you will not be downloading large volumes of data. As a final implication, its we are uncertain about how members of the amateur radio community would feel about using the APRS network this way.

In this final chapter, we will quickly review the key steps taken through the work. We will then see what contributions this thesis has made to the body of knowledge and the community.

### 6.1 Review

This thesis worked towards towards multiple linked objectives. Firstly we needed to determine if a successful downlink was possible between a typical CubeSat and the terrestrial APRS ground station network, not including interference. Secondly we needed to determine the impact of adding dispersed ground station receivers to a CubeSat’s downlink system as part of a virtualized ground station network. Thirdly we had to determine the capability boundaries of the virtualized ground station network on the common APRS frequency to a CubeSat telecommunications downlink system. Fourth we had to characterize some of the properties of received signals from a CubeSat to multiple ground stations.

Within this work we outlined the nature of the APRS I-Gate ground stations used within the network via a survey of the amateur radio community. From here a set of models of the antennas used within the ground station network were created. We also collected data from APRS network packet messages directly to determine realistic ground station locations.

A literature review determined the most popular CubeSat communications designs was combined with extracted data on most popular orbits and create a set of generalized CubeSat models. We then combined the ground station antenna models and CubeSat models into a comprehensive CubeSat communications system simulation. This allows us to generate representative received signals from the CubeSat to the ground stations.

This time series data set is then analyzed for its stationarity. If a unit root within the data

set indicates non-stationarity then common seasonality and trend stationarity methods can be used mitigate these factors. Next we will see how this work can be extended to answer more questions, paying particular attention to cognitive radio and by proposing a network design. From here we'll see what the limitations of this work are and what can be done to mitigate them. Finally the thesis will close with a few notes of personal experience and advice.

### 6.1.1 Considerations

Some key factors related to this work deserve highlighting as we consider its contributions.

The infrastructure of the APRS I-Gates already exists and is in use for day to day operations. Within North America over 842 are currently listening to 144.39 MHz for nearby nodes to beacon out packets. These nodes are heterogeneous in nature as volunteer amateur radio enthusiast build or buy their own equipment to participate within the network in much the same way as the conventional internet. Similar to the internet, the APRS network is decentralized with no particular authority controlling its use or operation beyond the established standards for interconnection.

In a typical AMSAT / OSCAR ground station a tracking antenna, with a directional main lobe far field antenna pattern, follows the satellite as it travels through its orbit. APRS nodes, including I-Gates, are typically designed terrestrial communications and typically use fixed antennas with omnidirectional far field patterns. None of the infrastructure, nor the frequency used, nor the modulation type, nor the protocol of AX.25, nor the APRS packet structure were created with space communications in mind. This thesis has proposed to this existing terrestrial infrastructure for CubeSat downlink communications.

## 6.2 Contributions

Throughout this thesis a number of to contributions are evident. Several of them are highlighted as follows:

1. This thesis shows that its possible for APRS I-Gate to correctly receive an AX.25, AFSK, 1200 baud packet from a Low Earth Orbiting satellite. Although projects like the FUNcube had physically demonstrated a related technique using their dongle, the process had not been explored using the APRS network or protocols. The thesis also experimented and documented what is required of a high altitude APRS beacon to prevent digipeter loops.
2. An unanticipated contribution of this work is that it uses what may be one of the most complete and largest scale RF network simulations to date. The simulation includes 3D far field radiation patterns for three of the most commonly used fixed station antennas from the amateur radio community. These far field patterns were chosen based on a survey of members of the amateur radio community. A by product of this effort was the importing and cataloging of these far field patterns into STK. Previously, the freely available antenna patterns were limited to commercial space application designs (1.2 GHz for example) or idealized dipoles and isotropic radiators rather than lower

frequency designs as used by the amateur radio community. The far field patterns were then distributed across a WGS84 oblate spheroid version of North America based on real world positions of APRS I-Gates.

3. A key contribution for this thesis is the characterization of packet reception in Sec. 5.3 for the I-Gate network from space. The equations within that section can enable a CubeSat communications designer to realistically determine the required  $E_b/N_0$  to achieve a desired BER if using APRS on their spacecraft. From there one can work backwards to determine the minimum transmit power and thus optimize their energy consumption. Remember that these equations are for the virtualized ground station network and not a typical AMSAT/OSCAR station design. However, the same methodology could be used to create the same equations for any satellite downlink using APRS.
4. The literature review for this thesis combined not only the history of CubeSat communications systems but details about their orbits as well. Previous studies had not incorporated the population orbital elements and thus had to make broad assumptions about CubeSat orbits. Furthermore, by including both the communications systems literature review and the population of CubeSat orbits this presents a set of generalized CubeSat models that can be utilized for further trade studies and engineering designs. This set of generalized CubeSat models can mitigate much of the estimations and guess work encountered by those new to the development of nanosatellites. As such the set of models limit the need for guessing the eventual orbit of a proposed CubeSat. The set of models can also enable more rapid analysis for link budgets and access times. By using STK rather than a customized software tool these generalized models can be made available for other CubeSat designers to jump start their development.
5. An enticing viewpoint on stationarity is that if the appropriate window size is chosen, a selection of the time series may appear stationary. Based on this concept analysis would be possible on non-stationary time series by optimizing the right window size, thus enabling the use of classic analytical methods such as Fourier transforms to characterize the system in question over that narrow selection of time. The thesis demonstrates, using practical tools, that this concept of analysis is problematic at best when attempting to use classical statistical calculations. At worst the method cannot distinguish between known stationary and known non-stationary time series and that the selection of short time window size has a more statistically significant impact than the series itself. Based on results from this thesis, further exploration of the Short Time Window Analysis of Non-Stationary Time Series method should be conducted by mathematicians and/or statisticians is a proof is possible.
6. While conducting the short time window analysis this thesis introduced the novel concept of the comparison of slopes as a measure of tendency across time for varying window sizes. This second order derivative is easy to calculate and plot may prove useful in other studies.
7. An interesting contribution of this work was the introduction of the non-parametric Phillips-Perron (PP) stationarity test which has seen very limited use outside the fields

of econometrics. The PP test was applied to known stationary White Noise and known non-stationary Brown Noise, both of which are familiar to engineering. In this way the thesis presented the utility of the PP test to the discipline of signals analysis in a way that is intuitive to understand and supported by evidence. The practicality of using the PP test as part of an embedded system for real time analysis remains to be explored. This thesis introduced the graphical analysis method PP p-value ratios as a method of evaluating the stationarity of a time series across a range of short time window sizes. This method could be used to evaluate the stationarity of a system such as altering a Kalman filter and checking that the output remains within the required limits.

8. The additional I-Gate nodes within the virtualized ground station network mitigate the changes in slant range induced spreading loss and cross-polarization problems due to non-alignment of ExampleSat's antennas with the ground station nodes. This enables ExampleSat to communicate with the virtualized ground station network despite not using any directional beam tracking antennas for the ground stations nor for the satellite itself. Not all of the accessible I-Gates will be at extreme slant ranges from ExampleSat at the same time, nor will they all be at extreme antenna cross-polarization angles at the same time.
9. By using the PP tests the thesis determined, beyond simple conjecture, that the aggregation of I-Gate nodes into a network creates a pooled opportunity for improved satellite access time and that the received signal can be treated as stationary. This includes improvements beyond those capable of a single ground station using a classic AMSAT/OSCAR Yagi-Uda directional antenna. The fact that the aggregated virtualized ground station network time series can be considered stationary (after  $\geq 250$  samples) implies that the full suite of traditional analytical and transformational methods of signals analysis and manipulation can now be brought to bear on the spacecraft communications system. This thesis represents what is effectually basic research on a satellite communications architecture that had not been studied before in open research.

## 6.3 Extensions To The Work

As with any level of research its always possible to explore the topic further. The simulation results used within the thesis were based only on ExampleSat-165 but additional orbital altitudes were determined and included within the greater scenario. These higher orbital altitudes could be used for a trade study to determine an optimization curve of orbital altitude to transmission power. This was initially avoided within the since CubeSat developers rarely determine the final orbit for their spacecraft as they often ride share into space. However, it still remains as a useful question to answer.

The AX.25 protocol and the APRS network are widely distributed and actively used around the world. Ideally a modification to the embedded messages that often company a typical beacon should include a measurement of the received signal strength for each packet. A combination of characters, often left unused, that expresses some measurement of the RF



channel that the packet transited through would enable Cognitive Radio applications as we will discuss in Sec. 6.3.2.

### 6.3.1 Amateur Radio Community Surveys

More extensive surveys of the amateur radio community could yield benefits such as a statistically representative sample of the antennas used by the population. Furthermore understanding the equipment used, at specified transmission power, at specified transmission intervals could lead to a wealth of data on radio interference based on patterns of life of the users. Such data would be useful not only for the community itself but also for radio manufacturers and other RF researchers.

### 6.3.2 I-Gates and Cognitive Radio

In their 2009 book “*Artificial intelligence in wireless communications*” [Rondeau2009], Thomas Rondeau and Charles Bostian credit Christian Rieser with coining the terms “knobs” and “meters” as applied to cognitive radio (CR) some time between 2001 and 2002. Regardless of when or whom created these terms, they are useful for our explanation of what is cognitive radio, and how can it be applied to our virtualized ground station network.

For a conventional radio the “knobs” are the adjustable controls the govern a radios operating parameters, while the “meters” display some measure of performance for the radio or the actual values of the operating parameters. “Knobs” can include the choice of modulation such as AM or FM, a transmission power of 5 W, or the transmission frequency of 100 MHz. “Meters” can be any form of displayed information about the radio system such as the received signal strength in dB (or bars in the case of current cell phones), an LED showing the signal is in stereo, or that the received signal has too many bit errors to be correctly interpreted. To quote Rondeau and Bostian,

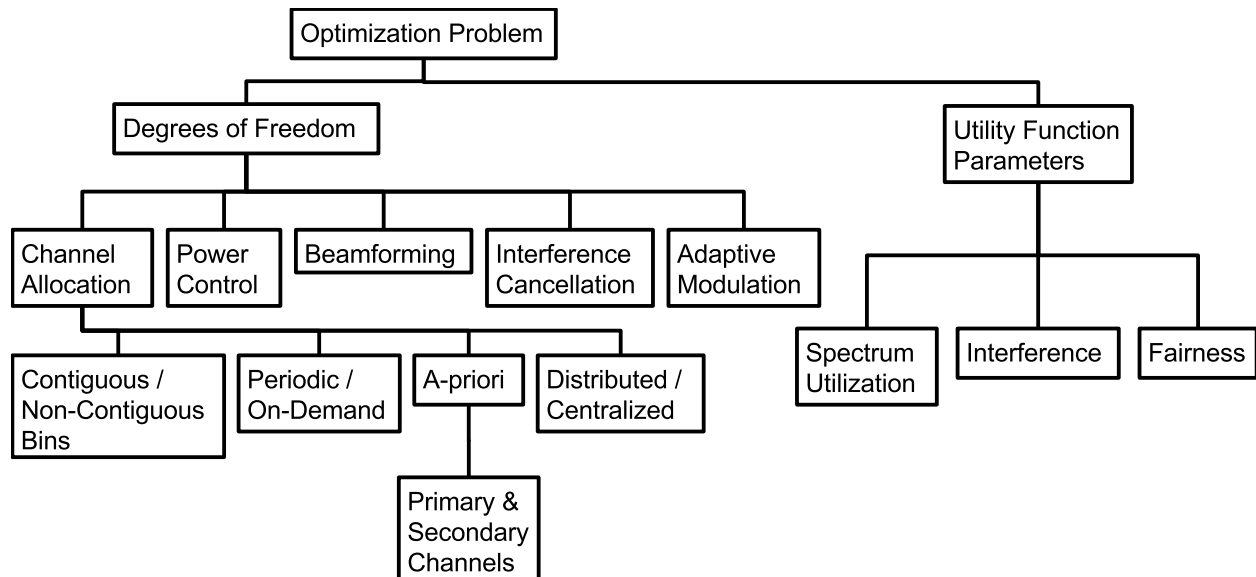
“...cognitive radio uses the meters to build an understanding of the environment so as to adjust the knobs to improve the communications.”

We can loosely define cognitive radio as an adaptive communications system that dynamically adjusts the parameters of the communications link to optimize that link in some predictive way. This optimization can focus on the limited resources available such as RF spectrum, modulated code space, transmission window time, signal phase or even polarization. However, many of these techniques may not be available for the virtualized ground station network, because we want to take advantage of existing infrastructure as an overlay network this implies many constraints on improving the overall systems design. Many of the “knobs” are not available for us to adjust and many of the “meters” are not shown to us as users of a virtualized ground station network.

#### 6.3.2.1 CR for the current network

George Fortetsanakis used a graphic similar to the one shown in Fig. 6.1 for his computer science Wireless and Mobile Computing lecture titled “*Introduction to Cognitive radios: Part one*” with the University of Crete [Fortetsanakis2011]. We will use this graphic in

conjunction with the “knobs” and “meters” principle to see what possible flavour of CR could be applied to the virtualized ground station network as is. For a CR system to function it must read the “meters” and then adjust the “knobs” as appropriate. Knowing which are available to the for design use will inform the policies that are implemented in CR system. Channel Allocation, Power Control, Beamforming, Interference Cancellation and Adaptive Modulation are examples of typical “knobs” available to CR designers. We shall begin from the perspective of an overlay network in which CRs check their meters that listen to and then enhance non-cognitive radio links.



**Fig. 6.1:** Cognitive Radio Problem Definition. After [Fortetsanakis2011]

Lets look at Channel Allocation first. As described in Sec. 2.5.3 the allocation of satellite downlink frequencies is globally coordinated for deconfliction via the ITU. This implies that dynamic frequency hopping would be limited to a fixed set if at all. Additionally, the APRS ground station I-Gate network nodes we want to use are tuned to a single frequency for a given geographic region. We have no control over the nodes directly, under this system design, we do not have access to their RF tuning frequency, this precludes using a Channel Allocation “knob” on the receiving end of the network.

We may be able to adjust the downlink frequency from the satellite to accommodate for Doppler shift, as was done by the OUFTI-1 CubeSat from Université de Liège [AMSATUK2013, Pisane2013] for their D-Star compatible spacecraft as described in Sec. 2.7. This technique would allow for the selection of specific ground stations be chosen for enabling a maximized downlink  $E_b/N_0$  compared to its peers. This link value would be determined by simulation in advance. Selected ground stations would be chosen in sequence to maximize both the received signal  $E_b/N_0$  but also the cumulative downlink time. To enable such a technique we would only need to modify the downlink transmitter’s TNC settings to enable the address field of the AX.25 packet. This would mean that any packet received by the APRS network would only be digitally repeated by the addressed node. This limits further traffic loading of the RF network.

The risk with this technique is based on the flaws in our network overlay design. Since we do not control the nodes we have no idea if the node would actually receive the  $E_b/N_0$  value that was calculated *a-prior*. The amateur radio operator could alter the hardware being used at any time. Remember that an individual operator or club owns each node, its access and its use. Based on these limitations its unlikely that we would want to use a Channel Allocation “knob” on the transmitting side of the link.

Power control would be of great interest for energy conservation onboard a CubeSat. By optimizing the transmit power from the satellite based on a meter reading of  $E_b/N_0$  at the ground station. The problem is that we do not actually have such a meter within the APRS I-Gate network. The radio equipment used may have such a measurement system but this information is not made available to the network at large. In fact, this is a key problem for any proposed meter within the current network architecture. It would be possible to build specialized nodes with this meter capability but that would defeat the purpose of the virtualized ground station network design of taking advantage of existing infrastructure.

Beamforming is the principle of dynamically adjusting the far field radiation or gain patterns of an antenna system to change the directionality of reception or transmission. This is often achieved through time phased arrays of antenna elements as with modern radar applications. The significant advantage of this technique is that it would allow for nulling of interference sources at the receiving end of the link (Interference Cancellation according to Fig. 6.1). From the transmitting end, on ExampleSat it would enable a stronger received signal to a narrower footprint on the Earth’s surface. Again, since we do not control the hardware in use with the I-Gates we could not use beamforming on the receiving end. Its possible that beamforming could be used onboard the satellite. This is an active area of research [Klein2014, Iwami2010] which so far includes lab prototypes.

Adaptive Modulation is also not promising “knob” for current network design. The APRS I-Gate nodes we want to use for our virtualized network also use a single modulation type at a specified baud. This implies that we cannot use different modulation types, or code division multiple access techniques to improve the link with these ground stations. This also precludes the use of differing signal phase methods as the AFSK modulation described in Sec. 2.4.2 implies that the phase remains continuous. Breaks in phase may introduce errors into the bit stream.

From examining the lack available “knobs” or “meters” that could be applied to the current network in an overlay capacity we are forced to look at future APRS network CR designs as an underlay network. A CR underlay network is constrained to cause minimal interference with non-CR radios.

### 6.3.2.2 CR for the future network

A detailed breakdown of the systems design and CR software requirements would be premature at this point of development but we can outline the key concepts and benefits of adding CR to a virtualized ground station network. As we determined in Sec. 6.3.2.1, there are few CR “knobs” or “meters” available for utilizing the current network design. However, with inexpensive Software Defined Radios (the NooElec SESDR SMARt is \$39.50 CDN) and single board computers (the Raspberry Pi 3 model B is \$46.16 CDN) such as those shown in Fig. 2.14a its possible to create a small form factor I-Gate that still services the needs of

the APRS community while also aiding in the development of CubeSat communications CR systems.

With some design changes to the SDR and the I-Gate node software these solutions could be used as an inexpensive underlay network. Because of the speed and/or multiprocessing capabilities of SDRs they can effectively monitor several frequencies or a band of frequency simultaneously as far as the user experience is concerned. They would be capable of receiving a relatively slow 1200 baud AX.25 packet on 144.39 MHz while also monitoring a second VHF or UHF frequency for a different modulation. This implies that an \$85.66 CDN radio system could perform as an APRS I-Gate as well as a CubeSat downlink station simultaneously based on CR principles.

As we mentioned at the beginning of Sec. 6.3.2 a “truly cognitive” CR system would have the ability to not only react and therefore adapt to the channel environment but to predict it and thus plan accordingly. The advantage of satellite communications from a CR perspective is that the LOS access times of the spacecraft are highly predictable. Their positions above the Earth are predetermined by orbit propagator algorithms and their position is also actively monitored by the Space Surveillance Network (SSN) radar stations. This implies that the predictive component for a CubeSat CR ground station has the added benefit of a-prior knowledge of which satellite may be transmitting to a receiving node at any given time. This limits the need for spectrum monitoring of the space side of the downlink which can be resource intensive in terms of time, energy and computation.

The SDR hardware would need to be altered such that it was capable of transmitting on 144.39 MHz to accommodate those amateur operators who wish to use the node not only as an I-Gate but as a digipeter as well. Other transmitting bands would be optional depending on demand or the CR network requirements. Reception capabilities for the SDR are really only limited by hardware and legality for a geographic area. Combined these transmission and reception capabilities would make such an SDR more akin to the Universal Software Radio Peripheral (USRP) or HackRF.

In effect, the principles that made the FUNcube dongle so popular could be further extended to provide a more ubiquitous service to the amateur radio community while at the same time providing a platform for the experimentation of CR in a practical environment. This proposed radio system would act as an I-Gate for the amateur radio community. During its normal operations the SDR would use spectrum sensing to determine the spectrum usage, interference and fairness of use of the allocated frequencies. From here it could implement any number of policies via its Utility Function Parameters. The APRS network would remain the primary use of such CR enabled I-Gate nodes, while the secondary purpose or a mitigated interference basis would be to conduct satellite communications.

Because of the frequency allocation for satellite communications via the ITU and due to the predictability of CubeSat orbits a centralized coordination of channel allocation of non-contiguous frequency bins is feasible and potentially easier to implement. Using a centralized virtualized ground station network of CR enabled nodes would permit capabilities such as space division multiplexing with satellites on opposite sides of the planet or with different footprints over the same continent. Time Division Multiplexing between satellites in close proximity would also be possible. Most significantly as demonstrated in Chapter 5 such a network could use statistical methods to compare received packets thus enabling Code Division Multiple Access where spread spectrum processing would provide much needed dB

gain via its improved signal-to-interference ratio. Weak signals with interference could take advantage of Viterbi decoding of convolutional or trellis codes for improved error correction. Belief vectors for algorithms could be augmented by measured packets across the network. Software and firmware updates could be pushed out or pulled down by I-Gate operators as new CR policies for spectrum use are developed. Furthermore, CR I-Gate nodes could further more fundamental CR research through spectrum sensing, interference measurements, experiments in Utility Function Parameters, A/B testing of spectrum policies and more. If such nodes include an accurate GPS they could also form an extremely large long baseline array for radio telescope based research.

There are several key advantages to enabling the amateur radio community with low priced CR enabled I-Gate nodes. As a community amateur radio operators are often interested in experimentation as well as new applications for their hobby. “Hams”, as they are colloquially called, are often considered early adopters of new technology. Thus they are more likely to want or acquire such a radio system. The 842 I-Gate nodes use in this thesis were collected from packets operated by real people using real hardware. The people operating these nodes already volunteer their hardware, software, expertise, electricity and space to enable the network to function. These I-Gate operating Hams at first glance would appear to be an ideal candidate group to participate in CR research. We should also consider that the names and address of the 842 volunteers are available through the call sign databases because Hams in North America are required to obtain a license to operate. As a research project it would cost \$72125 to purchase the basic hardware (without transmitter capabilities) for all 842 I-Gate operators, with mailing address that can be easily determined, to create a continent wide CR experimentation network. This would ensure wide dispersal across the continent. Such a research project is within the realm of the possible for a well funded university or government institution.

## 6.4 Limitations and Potential Solutions

As described in Sec.4.5.2.1 there are limitations to the models and the simulations used throughout this thesis. To begin we must acknowledge again the words of George E.P. Box “Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful.” To that end the models and simulation have proven useful for exploring the proposed solution space.

More far field antenna models for both ground stations and CubeSats were created than were ultimately used within the simulations. In part this was to limit the scope of the work. Its was also due to the limited number of responses to the amateur radio community antenna survey which were not statistically representative of the population. The requirement for ethical standards in research involving human subjects participating in an online survey needs to be streamlined and more responsive. A larger social media strategy is required for this type of research and is not typical within engineering training.

The modulation type used within the scenario was FSK with equidistant 7 kHz bandwidth limits to emulate AFSK within the discrete event simulation environment. Ideally a customized modulation script would be used but would be very difficult to write as AFSK is a form of technological bridge between older analog radio systems and modern digital

techniques.

The impact of Doppler Shift and Doppler Rate on the reception of digital packets remains an open question. The planned satellite channel emulation experiment in Sec. 3.4 was not possible at the time of writing. We will have to wait on further software development of the GNU Radio platform to continue this aspect of the research. In particular the documentation of the software for those outside of its development group needs improvement. The results used in this work focused on a single orbit of 165 km in altitude. This was initially chosen as a “worst case scenario” with respect to the impact of Doppler Shift and Doppler Rate to coincide with the proposed emulation experiment. By the time it was discovered that the emulation experiment would not be feasible it was too late to reselect the orbital altitude. However, these altitudes are still available within the simulation set for further study. In addition the selected altitude of 165 km still contains a large enough population of CubeSats to remain informative. The net impact of different altitudes will be changes in Doppler Shift, Doppler Rate, coverage area or footprint, Rx  $E_b/N_0$ , and LOS access duration.

The set of STK simulations is limited in that does not emulate or calculate a network traffic, loading or delay as would network simulation software such as ns-3 or OMNET++. Such a simulation would require creating state agents/machines of the AX.25 protocol as these are not currently available for the typical network simulation software tools.

At this time it is not recommended that a CubeSat mission designer build and launch a spacecraft that uses APRS I-Gate nodes directly. Because of the coverage area accessible by a CubeSat from its orbit a large number of nodes could be interfered with during the spacecraft’s transmissions. Although the APRS network inherently self interfering due to the random nature of each node’s beacon interval, the level of interference caused by transmitting from a satellite is unknown at this time. It is easy to assume that since a typical CubeSat uses a 1W transmitter, that local nodes which use higher power (45% of 104 surveyed operators use between 1W and 5W for their mobile nodes) and are much closer they would overcome the satellite’s signal. Additionally, since the CubeSat would use packets without an specified address, packets would only be digipeted by I-Gate nodes to the internet, preventing “flooding” of the network. However, extensive modelling is required to determine a realistic level of interference. Trepidation and concern was expressed by members of the amateur radio APRS community during informal discussions about this research. Consultation with the APRS stakeholder community must occur before any widespread use of the network for space systems communications.

## 6.5 Personal

From a personal perspective, doing this research has re-kindled my interest in amateur radio and other RF technologies. Growing up in a household prior to the ubiquity of cellular phones but surrounded by amateur radio, the hobby was initially just a means of communication for me. It worked because the only other alternatives were twisted pair land line phones or physical mail. As a kid, I remember sending a short hello message via packet radio teletype to a man in Russia before Glasnost; while the Iron Curtain still obscured our world view. Sadly I do not remember his call sign.

With the advent of email and mobile communications that enabled us to carry the internet

in our pocket, amateur radio seemed antiquated and somewhat unneeded. But as open source platforms such as Arduino make electronics and software experimentation accessible to the masses at low cost its not possible to create things that were previously not possible with amateur radio. It's now possible to use an off the shelf amateur radio Arduino transceiver shield and hack together a near space weather balloon experiment or a packet radio enabled garage door opener (not recommended for security reasons).

My interest space systems remains. I briefly put this thesis on hiatus while acting as project manager for the UMSATS team to create our own CubeSat as part the Canadian Satellite Design Challenge[Schor2013]. This process was deeply frustrating, exhausting, and entirely fantastic. Through it I met extraordinary people learned more than I could in any classroom alone.

## Advice for other students

I initially started my graduate studies for a deeply personal reason. I wanted to prove to myself that I could do it. Despite my academic history, despite my shortcomings. What did I learn? I was wrong about my motivations. I did it because I needed to know the answers to “Can it be done? Would it work?” Do not do STEM based graduate studies part time. It takes more time and effort than anyone can reasonably afford. Universities are not set up for adult students. Professors and classmates do not quite understand that you have a career and other commitments outside of the university. “No I can not leave work and come to a meeting across town in half an hour.” Save yourself the trouble, and figure out a way to do it full time. You'll thank yourself later.

And finally: Do graduate studies. What have you got to lose? Other than your money, you free time, and your sense of perspective. What you'll gain is not a piece of paper or a qualification. What you'll gain is a slice expertise and deeper knowledge of yourself, more than any text book or classroom subject. You'll be qualified for a life of learning. How cool is that?!





# References

- [AGI2016a] AGI, “External Antenna Pattern Files,” help files, 2016, URL <http://help.agi.com/stk/Content/comm/CommRadarB-01.htm>.
- [AGI2016] AGI, “STK,” website, 2016, URL <http://www.agi.com/products/stk/>.
- [AGI2016b] AGI, “STK Analyzer,” website, 2016, URL <http://www.agi.com/products/stk/modules/default.aspx/id/stk-analyzer>.
- [AGI2016c] AGI, “STK Optimizer,” website, 2016, URL <http://www.agi.com/products/stk/modules/default.aspx/id/stk-optimizer>.
- [a.i.solutions2016] a.i. solutions, “FreeFlyer,” website, 2016, URL <https://ai-solutions.com/freelyer/freelyer/>.
- [Alexander2013] S. Alexander and R. D. Marlaire, “NASA Successfully Launches Three Smartphone Satellites,” website, 2013, URL <http://www.webcitation.org/6YALok8vZ>, link via web archive.
- [Ali1998a] I. Ali, N. Al-Dhahir, and J. Hershey, “Doppler characterization for LEO satellites,” *Communications, IEEE Transactions on*, vol. 46(3), pp. 309–313, 1998.
- [Allaire2016] J. Allaire, “R Studio,” website, 2016, URL <https://www.rstudio.com/>.
- [AMSAT2016] AMSAT, “ASMAT,” website, 2016, URL <http://www.amsat.org/>.
- [AMSATUK2013] AMSATUK, “Integration of Worlds First D STAR Cube-Sat,” website, 2013, URL <https://amsat-uk.org/2013/10/22/integration-of-worlds-first-d-star-cubesat>.
- [Analytics2016] R. Analytics, “inside-R,” website, 2016, URL <http://www.inside-r.org/>.
- [APRS.fi2016] APRS.fi, “aprs.fi Application Programming Interface · Google Maps APRS,” website, 2016, URL <http://aprs.fi/page/api>.
- [Arnold2012] S. S. Arnold, R. Nuzzaci, and A. Gordon-Ross, “Energy budgeting for CubeSats with an integrated FPGA,” in *Aerospace Conference, 2012 IEEE*, pp. 1–14, IEEE, 2012.

- [Banerjee1993] A. Banerjee, J. J. Dolado, J. W. Galbraith, D. Hendry, *et al.*, “Co-integration, error correction, and the econometric analysis of non-stationary data,” *OUP Catalogue*, 1993.
- [Barker2016] R. Barker, “The Official UI-View32 Web Site,” Website, 2016, the original software was written by Roger Barker (G4IDE). The website has been updated since G4IDE’s key went silent.
- [BBN1996] BBN, “PROPHET StatGuide: Do your data violate one-way ANOVA assumptions?” website, 1996, URL [http://www.basic.northwestern.edu/statguidefiles/oneway\\_anova\\_alts.html#Transformations](http://www.basic.northwestern.edu/statguidefiles/oneway_anova_alts.html#Transformations).
- [Bedon2010] H. Bedon, C. Negron, J. Llantoy, C. Nieto, and C. Asma, “Preliminary internetworking simulation of the QB50 cubesat constellation,” in *Communications (LATINCOM), 2010 IEEE Latin-American Conference on*, pp. 1–6, 2010.
- [Beech1998] W. Beech, D. Nielsen, and J. Taylor, “AX.25 Link Access Protocol for Amateur Packet Radio,” Tech. rep., Tucson Amateur Packet Radio, 1998.
- [Bethea1991] R. M. Bethea and R. R. Rhinehart, *Applied engineering statistics*, vol. 121, CRC Press, 1991.
- [Boehm1989] B. Boehm, “Software risk management,” in *European Software Engineering Conference*, pp. 1–19, Springer, 1989.
- [Boithias1987] L. Boithias, *Radio Wave Propagation*, North Oxford Academic, 1987, ISBN 9780946536061.
- [Bouwmeester2010] J. Bouwmeester and J. Guo, “Survey of worldwide pico-and nanosatellite missions, distributions and subsystem technology,” *Acta Astronautica*, vol. 67(7), pp. 854–862, 2010.
- [Bunescu2013] R. Bunescu, N. Struble, C. Marling, J. Shubrook, and F. Schwartz, “Blood Glucose Level Prediction Using Physiological Models and Support Vector Regression,” in *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, vol. 1, pp. 135–140, 2013.
- [Buscher2014] M. Buscher, “TUB Small Satellite Database,” Spreadsheet, 2014, URL <http://www.raumfahrttechnik.tu-berlin.de/fileadmin/fg169/Projects/REPIN/TUBSmallSatelliteDatabase.xlsx>.
- [Castello2012] B. R. Castello, *CubeSat Mission Planning Tool*, Master’s thesis, California Polytechnic Stat University, San Luis Obispo, 2012.

- [CRAN2016] CRAN, “The Comprehensive R Archive Network,” website, 2016, URL <https://cran.r-project.org/>.
- [Diaz2016] M. Diaz, “International Space Hall of Fame,” website, 2016, URL <http://www.nmspacemuseum.org/halloffame/detail.php?id=29>, quote of Wernher von Braun, cited by New Mexico Department of Cultural Affairs.
- [Dickey1979] D. A. Dickey and W. A. Fuller, “Distribution of the estimators for autoregressive time series with a unit root,” *Journal of the American statistical association*, vol. 74(366a), pp. 427–431, 1979.
- [Dimse2014] S. Dimse, “personal communications regarding FindU.com, q-constructs, and APRS in general.”, 2014, special Thanks to K4HG.
- [Finnegan2014] K. W. Finnegan, *Examining Ambiguities in the Automatic Packet Reporting System*, Master’s thesis, California Polytechnic State University San Luis Obispo, 2014.
- [Fortetsanakis2011] G. Fortetsanakis, “Introduction to Cognitive radios: Part one,” Slides, 2011, URL <http://www.csd.uoc.gr/~hy439/schedule.html>, part of CS439 lecture set: Wireless Networks and Mobile Computing.
- [Foundation2016] R. P. Foundation, “RASPBERY PI 1 MODEL B+,” website, 2016, URL <https://www.raspberrypi.org/products/model-b-plus/>.
- [Frenzel2015] L. Frenzel, “Whatever Happened to WiMAX?” website, 2015, URL <http://mwrf.com/blog/whatever-happened-wimax>.
- [Gagliardi1975] U. Gagliardi, “Trends in computing-system architecture,” *Proceedings of the IEEE*, vol. 63(6), pp. 858–862, 1975.
- [Garratt2013] A. Garratt, “WSPR using a FUNCube Dongle PRO+,” blog, 2013, URL <http://nerdsville.blogspot.ca/2013/03/wspr-using-funcube-dongle-pro.html>.
- [Gatward2009] G. Gatward, “APRS-IS Structure,” , 2009, <http://www.aprs.net.au/internet/aprs-internet>.
- [Goode1984] S. Goode, “BER Performance of TAPR TNC Modem,” Tech. rep., Tucson Amateur Packet Radio, 1984.
- [Heidt2000] H. Heidt, J. Puig-Suari, A. Moore, S. Nakasuka, and R. Twiggs, “CubeSat: A new generation of picosatellite for education and industry low-cost space experimentation,” in *14TH 1 Annual/USU Conference on Small Satellites*, 2000.
- [Hoots1988] F. Hoots, R. Roehrich, and T. Kelso, “Models for Propagation of NORAD Element Sets,” , 1988, URL <https://celestrak.com/NORAD/documentation/spacetrk.pdf>.

- [Hubing2008] T. Hubing, C. Su, Z. H., and K. H., “Survey of Current Computational Electromagnetics Techniques and Software,” Tech. rep., Clemson University Vehicular Electronics Laboratory, 2008.
- [Huiskamp2012] W. Huiskamp, J. Igarza, and A. Voiculet, “NATO Modelling and Simulation Standards Profile,” Tech. rep., NATO Standardization Agency (NSA), 2012.
- [Ibanez2012] I. Ibáñez, “Notes on antenna simulation using NEC-2,” Unión de Radioaficionados Españoles article, 2012, URL <http://www.ipellejero.es/tecnico/nec-2/english.php#2>, translation of original article.
- [IEEE2009] IEEE, “802.16-2009 - IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems,” , 2009, superseded by 802.16-2012.
- [IGPP2014] IGPP, “ELFIN Communications,” webstite, 2014, URL <http://elfin.igpp.ucla.edu/spacecraft-subsystems/comm/>, 4Nec2 files.
- [Ihaka2016] R. Ihaka and R. Gentleman, “The R Project for Statistical Computing,” website, 2016, URL <https://www.r-project.org/>.
- [ITU2001] ITU, “RECOMMENDATION ITU-R P.618-7 Propagation data and prediction methods required for the design of Earth-space telecommunication systems,” Tech. rep., ITU, 2001.
- [ITU2012] ITU, “Amateur-Satellite Service,” PDF, 2012, URL <http://www.itu.int/en/ITU-R/space/AmateurDoc/AmateurSatServiceFreq.pdf>.
- [ITU2012a] ITU, “Frequency allocations of amateur satellite service,” PDF, 2012, URL [http://www.itu.int/en/ITU-R/space/AmateurDoc/ARS-ART5\\_E.pdf](http://www.itu.int/en/ITU-R/space/AmateurDoc/ARS-ART5_E.pdf).
- [ITU2012b] ITU, “Resolution Com6/10 (WRC-12); Regulatory aspects for nano- and picosatellites,” in *The World Radiocommunication Conference (Geneva)*, International Telecommunications Union, 2012.
- [ITU2013] ITU, “Attenuation by atmospheric gases,” Tech. rep., ITU, 2013.
- [Iwami2010] R. T. Iwami, B. J. Lei, T. Lim, W. G. Tonaki, T. F. Chun, A. Zamora, J. C. Longhi, M. K. Watanabe, J. M. Akagi, and W. A. Shiroma, “Miniaturization challenges for autonomously phased arrays in a Cube-Sat application,” in *Wireless Information Technology and Systems (ICWITS), 2010 IEEE International Conference on*, pp. 1–4, 2010.
- [JitterLabs2015] JitterLabs, “BER Confidence-level Calculator,” website, 2015, URL <https://www.jitterlabs.com/support/calculators/ber-confidence-level-calculator/>.

- [Kelso2015] T. Kelso, “CubeSat TLE Query,” website, 2015, URL <http://celestrak.com/NORAD/elements/cubesat.txt>, queries the JFCC SPACE/J3 TLE database.
- [King2008] J. A. King, “AMSAT IARU Standard Link Budget System,” MS Excel spreadsheet, 2008, URL <http://www.amsatuk.me.uk/iaru/spreadsheet.htm>, w3GEY.
- [King2010] J. A. King, “AMSAT-IARU Basic Analog Transponder Link Budget Rev1.6,” spreadsheet, 2010, URL <http://www.amsatuk.me.uk/iaru/spreadsheet.htm>.
- [Klein2014] J. Klein, J. Hawkins, and D. Thorsen, “Improving cubesat downlink capacity with active phased array antennas,” in *2014 IEEE Aerospace Conference*, pp. 1–8, 2014.
- [Klofas2008] B. Klofas and J. Anderson, “A Survey of CubeSat Communication Systems,” Tech. rep., California Polytechnic State University SRI International, 2008.
- [Klofas2013] B. Klofas and K. Leveque, “A Survey of CubeSat Communication Systems 2009 to 2012,” in *CubeSat Developers Workshop*, 2013.
- [LeRoux2014] J.-H. Le Roux, *Development of a satellite network simulator tool and simulation of AX. 25, FX. 25 and a hybrid protocol for nano-satellite communications*, Ph.D. thesis, Stellenbosch: Stellenbosch University, 2014.
- [Lee2015] S. Lee, A. Hutputanasin, A. Toorian, W. Lan, R. Munakata, J. Carnahan, D. Pignatelli, and A. Mehrparvar, “CubeSatat design specification Rev. 13,” *The CubeSat Program, California Polytechnic State University*, vol. 1, pp. 1–13, 2015.
- [Leveque2007] K. Leveque, J. Puig-Suari, and C. Turner, “Global Educational Network for Satellite Operations (GENSO),” , 2007.
- [Liddle2011] D. Liddle, B. Dyer, J. Parsons, D. Feltham, R. Taylor, D. Mellor, A. Schofield, R. Linehan, R. Long, J. Fernandez, H. Kadhem, P. Davies, J. Gebbie, N. Holt, P. Shaw, L. Visagie, T. Theodorou, V. Lappas, and C. Underwood, “STRaND-1 Use of a 500 dollar smartphone as the central avionics of a nanosatellite,” in *62nd International Astronautical Congress*, 2011.
- [Long2011] H. Long and R. Peggram, “The FUNcube Dongle An Introduction,” PDF, 2011, g6LVB and G7RUH.
- [Loveall2005] P. Loveall, “How APRS Works,” Presentation, 2005.

- [Loveall2016a] P. Loveall, “APRS-IS,” website, 2016, URL <http://aprs-is.net/>, retrieved 27-04-2016.
- [Loveall2016] P. Loveall, “q Construct,” website, 2016, URL <http://www.aprs-is.net/q.aspx>.
- [Lustig2015] M. Lustig, F. Ong, and G. Fanti, “Lab 6 Part B AX.25 and APRS,” course manual, 2015, URL [https://inst.eecs.berkeley.edu/~ee123/sp15/lab/lab6/Lab6\\_Part\\_B-APRS.html](https://inst.eecs.berkeley.edu/~ee123/sp15/lab/lab6/Lab6_Part_B-APRS.html).
- [Masliah2002] M. R. Masliah, “Stationarity and non-stationarity identification,” *Internet Publication*, 2002.
- [Miller1988] J. Miller, “9600 Baud packet radio modem design,” in *ARRL 7th Computer Networking Conference*, pp. 135–140, 1988.
- [Miller2016] S. Miller, “1200 Baud Packet Radio Details,” website, 2016, URL [N1VG.net](http://N1VG.net).
- [Mills2016] C. Mills, “Xastir APRS Program,” website, 2016, curtis Mills (WE7U) was the most recent editor of the wiki page.
- [Minoli2009] D. Minoli, *Satellite systems engineering in an IPv6 environment*, Auerbach Publications, 2009.
- [Miura2009] N. Z. Miura, *Comparison and design of Simplified General Perturbation Models (SGP4) and code for NASA Johnson Space Center, Orbital debris program office*, Master’s thesis, California Polytechnic State University San Luis Obispo, 2009.
- [Morg2013] D. Morgan, “Further Developments of an SDR Radio Telescope,” British Astronomical Association, 2013.
- [Muenchen2015] B. Muenchen, “Rexer Analytics Survey Results,” wordpress, 2015, URL <http://r4stats.com/2015/10/13/rexer-analytics-survey-results/>.
- [NanoRacks2016] NanoRacks, “SmaSmall Deployment,” Website, 2016, URL <http://nanoracks.com/products/smallsat-deployment/>.
- [NASA2011] NASA, “Definition of Two-line Element Set Coordinate System,” website, 2011, URL [http://spaceflight.nasa.gov/realdata/sightings/SSapplications/Post/JavaSSOP/SSOP\\_Help/tle\\_def.html](http://spaceflight.nasa.gov/realdata/sightings/SSapplications/Post/JavaSSOP/SSOP_Help/tle_def.html), retrieved 23 Apr. 2016.
- [SCaN2014] S. C. Navigation, “Near Earth Network,” website, 2014, URL <https://goo.gl/boIAhW>.

- [NCS1997] NCS, “Telecommunicaitons Glossary of Telecommunications Terms,” , 1997, URL <http://www.its.bldrdoc.gov/fs-1037/fs-1037c.htm>, federal Standard 1037C.
- [NIMA2000] NIMA, “NGA: DoD World Geodetic System 1984,” , 2000, URL <http://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf>.
- [HackRFOne] M. Ossmann, “HackRF One,” website, 2016, URL <https://greatscottgadgets.com/hackrf/>.
- [Pacier2016] P. Pacier and H. Golding, “The Tier 2 Network,” website, 2016, URL <http://www.aprs2.net/t2network.php>, accessed 04-16-2016.
- [Paffett1998] J. Paffett, T. Jeans, and J. Ward, “VHF-band interference avoidance for next-generation small satellites,” in *12th AIAA USU Conference on Small Satellites*, AIAA/USU, 1998, 12th AIAA/USU Conference on Small Satellites.
- [Phillips1988] P. C. Phillips and P. Perron, “Testing for a unit root in time series regression,” *Biometrika*, vol. 75(2), pp. 335–346, 1988.
- [Pisane2013] J. Pisane, A. Denis, and J. Verly, “D-STAR digital amateur communications in space with OUFTI-1 CubeSat,” newsletter number 202, 2013, URL <http://www.passion-radio.org/images-blog/amsatuk-oufti-dstar-satellite.pdf>.
- [Qarabaqi2014] P. Qarabaqi, “Statistical characterization of a class of underwater acoustic communication channels,” *Electrical Engineering Dissertations, paper*, vol. 106, 2014.
- [QB502016] QB50, “Mission Objectives,” website, 2016, URL <https://www.qb50.eu/index.php/project-description-obj/mission-objectives>.
- [GNURadio2016] G. Radio, “GNU Radio; The Free and open software radio ecosystem,” website, 2016, URL <http://gnuradio.org/>.
- [RF2013] A. RF, “Link Budget Analysis: Digital Modulation, Part 2,” Presentation, 2013, URL <http://www.atlantarf.com/Downloads.php>.
- [Rondeau2009] T. W. Rondeau and C. W. Bostian, *Artificial intelligence in wireless communications*, Artech House, 2009.
- [Schor2013] D. Schor, G. Linton, and W. Kinsner, “Design, Implementation, and Testing of the T-Sat1 Nanosatellite at the University of Manitoba,” in *Proc. of the 64th International Astronautical Congress; IAC2013*, pp. 1–7, International Astronautical Congress, 2013.

- [Schroer2009] M. Schroer, “NPS-SCAT: A CubeSat Communications System Design, Test, and Integration,” Tech. rep., DTIC Document, 2009.
- [Shapiro1965] S. S. Shapiro and M. B. Wilk, “An analysis of variance test for normality (complete samples),” *Biometrika*, vol. 52(3/4), pp. 591–611, 1965.
- [Shirville2016] G. Shirville, “Introduction to FUNcube,” Website, 2016, URL ["funcube.org.uk"](http://funcube.org.uk).
- [Shirville2007] G. Shirville and B. Klofas, “Genso: A global ground station network,” in *AMSAT Symposium*, 2007.
- [Shtun1995] D. Shtun, “Radio Amateurs of Canada Band Plans,” web site, 1995, URL <http://www.rac.ca/en/>.
- [Sitaraman2014] R. K. Sitaraman, M. Kasbekar, W. Lichtenstein, and M. Jain, “Overlay networks: An akamai perspective,” *Advanced Content Delivery, Streaming, and Cloud Services*. John Wiley Sons, 2014.
- [smith2012r] D. Smith, “R tops data mining software poll,” *Java Developers Journal*, 2012.
- [Sokur2012] T. Sokura, “Ham-APRS-FAP,” The Comprehensive Perl Archive Network, 2012, URL <http://search.cpan.org/dist/Ham-APRS-FAP/FAP.pm>.
- [SPRSA2016] SPRSA, “The 18th Annual Small Payload Rideshare Symposium,” website, 2016, URL <https://www.sprsa.org/>.
- [Srivastav2015] T. Srivastava, “A Complete Tutorial on Time Series Modeling in R,” Website, 2015, URL <http://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/>.
- [Swartwout2012] M. Swartwout, “A statistical survey of rideshares (and attack of the CubeSats, part deux),” in *Aerospace Conference, 2012 IEEE*, pp. 1–7, 2012.
- [Tabakovic2000] Ž. Tabaković, “Doppler effect in non-GSO satellite propagation,” in *Millennium conference on antennas and propagation-IEEE AP-2000*, Hrvatska znanstvena bibliografija i MZOS-Svibor, 2000.
- [Taylor2014] R. Taylor, “Mini-satellites send high-definition views of Earth,” website, 2014, URL <http://www.bbc.co.uk/news/technology-27408318>.
- [Thakker2010] P. Thakker and W. Shiroma, *Emergence of pico-and nanosatellites for atmospheric research and technology testing*, American Institute of Aeronautics and Astronautics, 2010.



- [Thieler2016] A. Thieler, R. Fried, and J. Rathjens, “RobPer: An R Package to Calculate Periodograms for Light Curves Based on Robust Regression,” *Journal of Statistical Software*, vol. 69(9), pp. 1–36, 2016.
- [Thuvt2011] Thuvt, “CubeSat Flight Model.jpg,” Wikimedia Commons image, 2011, URL [https://commons.wikimedia.org/wiki/File:F-1\\_CubeSat\\_Flight\\_Model.jpg](https://commons.wikimedia.org/wiki/File:F-1_CubeSat_Flight_Model.jpg).
- [Trapletti2016] A. Trapletti, K. Hornik, and B. LeBaron, “tseries: Time Series Analysis and Computational Finance,” , 2016, URL <https://cran.r-project.org/web/packages/tseries/tseries.pdf>, version 0.10-35.
- [Voors2016] A. Voors, “4nec2,” website, 2016, URL <http://www.qsl.net/4nec2/>.
- [Wall2013] M. Wall, “Planet Labs Unveils Tiny Earth-Observation Satellite Family,” website, 2013, URL <http://www.space.com/22622-planet-labs-dove-satellite-photos.html>.
- [Wallio2015] R. Wallio, “AX.25 Packet Radio AFSK on FM Bit, Packet and Data Set Error Rates,” website, 2015, URL [http://showcase.netins.net/web/wallio/BER\\_Packetradiobiterrorate.html](http://showcase.netins.net/web/wallio/BER_Packetradiobiterrorate.html).
- [Weggelaar2006] W. Weggelaar and S. Speretta, “06Weggelaar.pdf,” Presentation Slides, 2006, URL [https://www.qb50.eu/download/3rdQB50Workshop\\_presentations/06Weggelaar.pdf](https://www.qb50.eu/download/3rdQB50Workshop_presentations/06Weggelaar.pdf), qB50 communications.
- [Wen2013] H. Wen, P. K. Tiwary, and T. Le-Ngoc, *Wireless Virtualization*, Springer, 2013.
- [Whatton] P. Whatton, “The Funcube Dongle for 2m DX.” website, URL [http://www.g4dcv.co.uk/2msdr/FCD\\_G4DCV.htm](http://www.g4dcv.co.uk/2msdr/FCD_G4DCV.htm).
- [WileyJ.Larson2004] J. R. W. Wiley J. Larson, editor, *Space Mission Analysis and Design*, Space Technology Library, Kluwer Academic Publishers, 3rd edn., 2004.



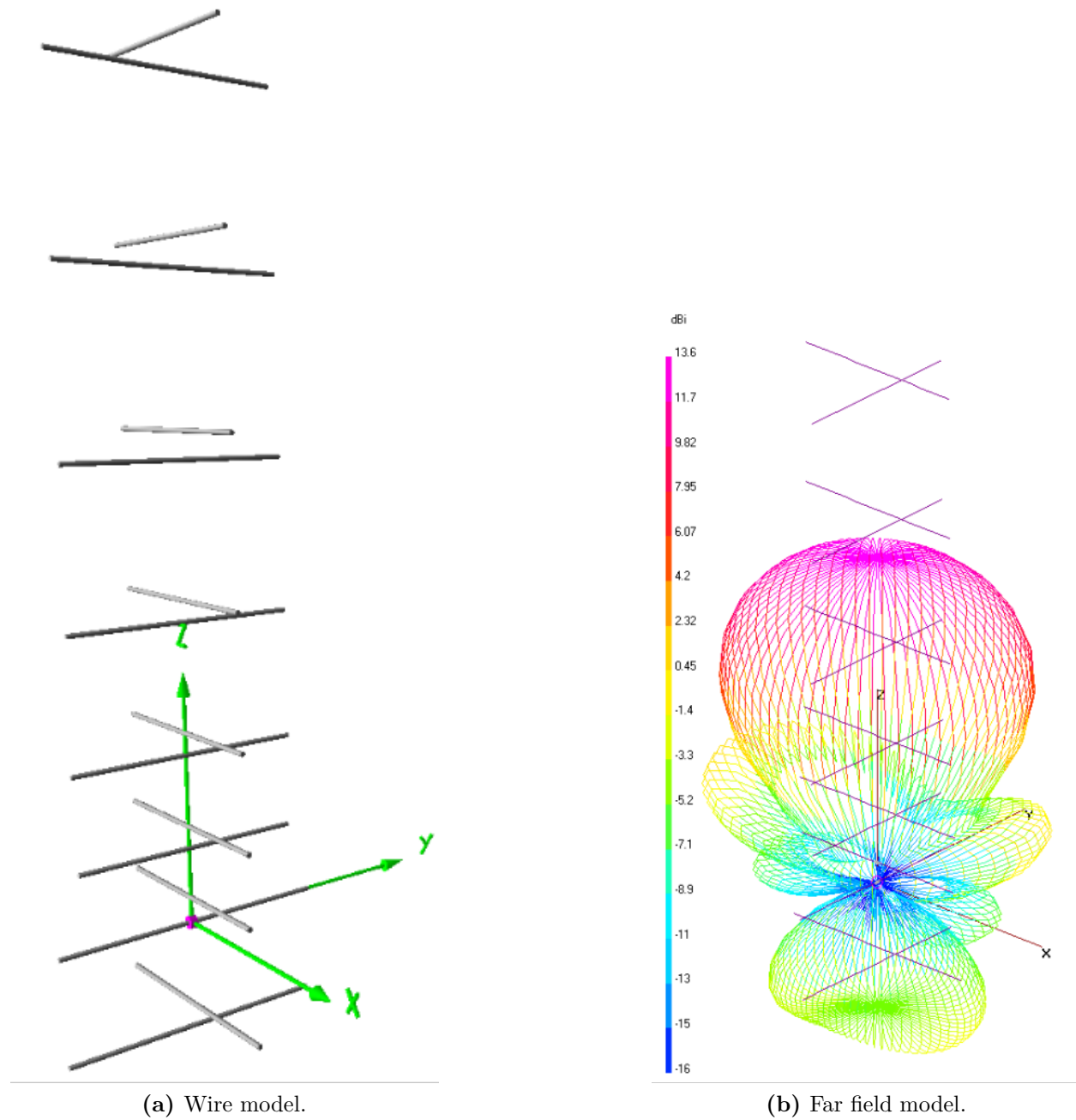
# A Appendix: Modelling

A PDF copy of this thesis along with the various simulations, models and data files are available via [www.gregjlinton.com](http://www.gregjlinton.com).

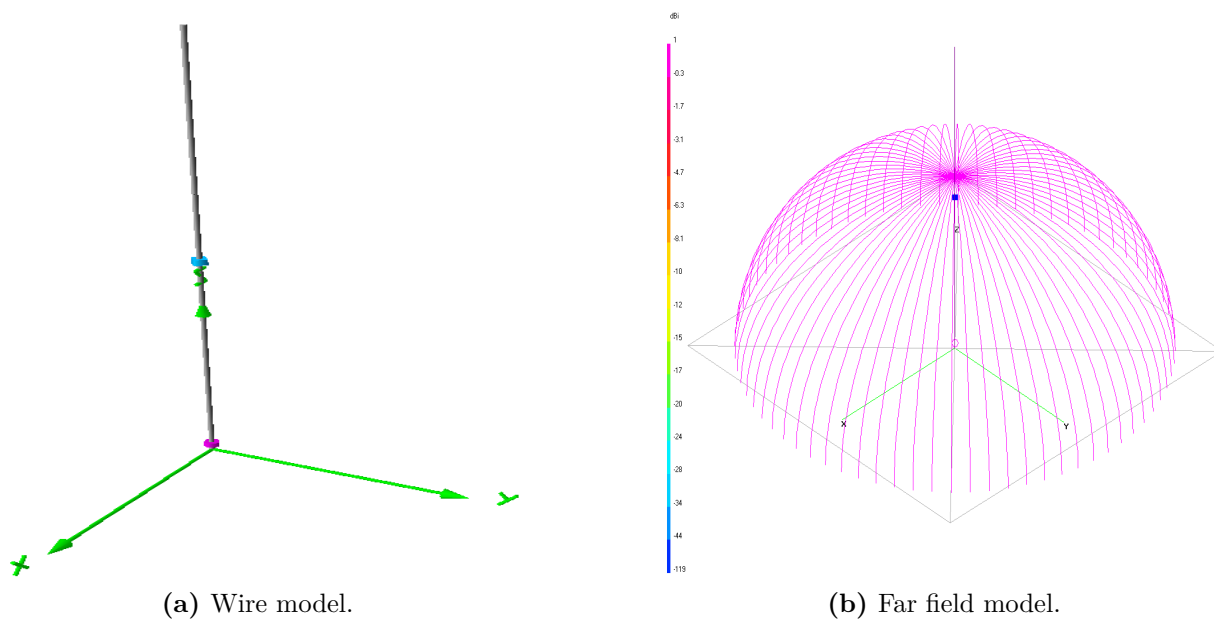
## A.1 Ground Station Antenna models

- GP144-435
- 2mGP
- J-ant144
- 216CP\_SatComMod

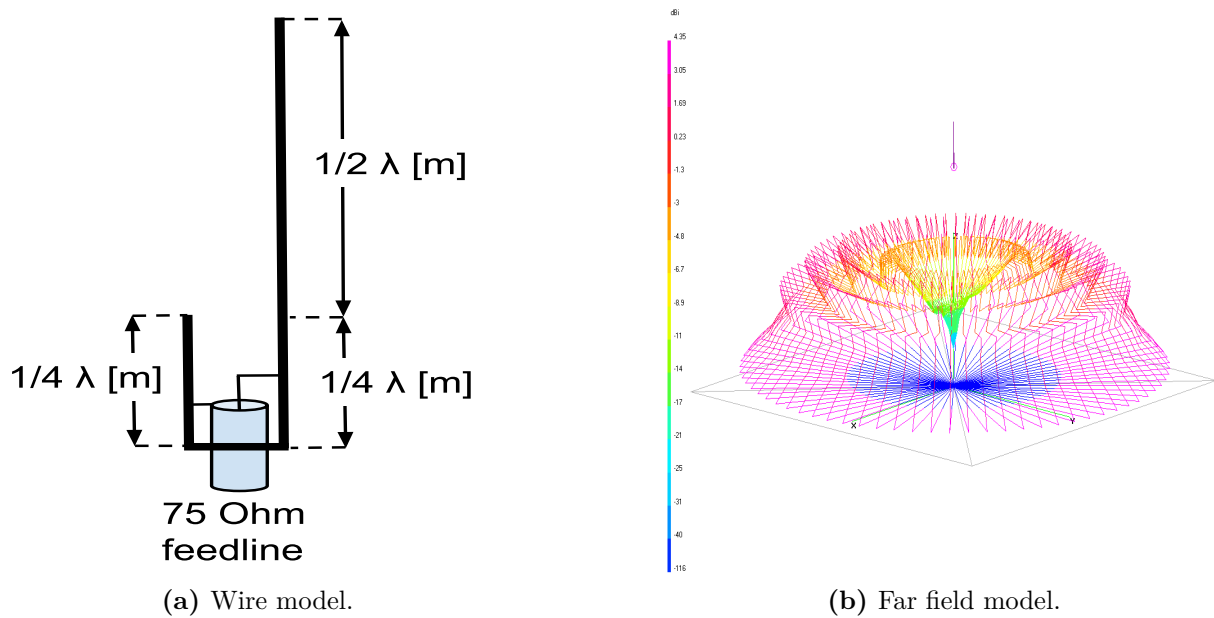
The 216+CP\_SatComMod is a 16 element Yagi-Uda design for the 2 m band. It has a single feed point to one element and can achieve a gain of 13.8 dB at the centre of its main lobe. This antenna is a modification of the 216CP two meter band experimental design that is included with 4Nec2. The modification involved adding a second set of orthogonal elements at an offset to enable differing polarizations within interference between the elements.



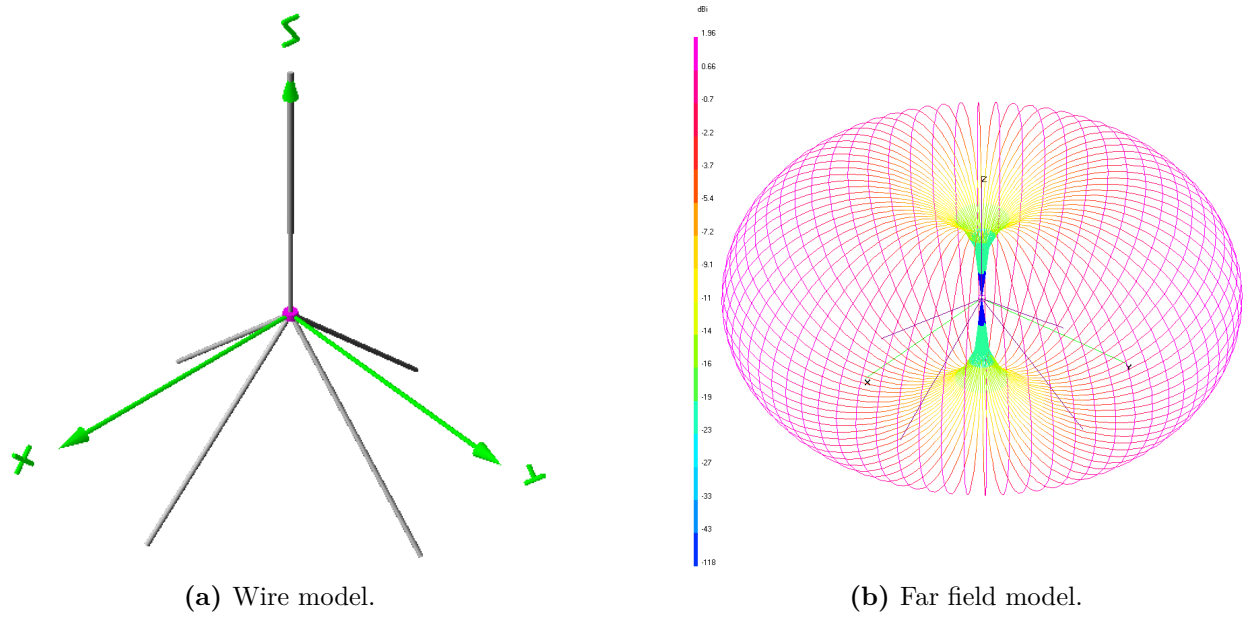
**Fig. A.1:** VE4UM directional station antenna model in 4Nec2.



**Fig. A.2:** GP144-435 omnidirectional antenna model in 4Nec2.



**Fig. A.3:** J-Ant144 omnidirectional antenna model in 4Nec2.



**Fig. A.4:** 2mGP omnidirectional antenna model in 4Nec2.

## A.2 Orbital Mechanics

### A.3 Parameters Required to Analyze the Downlinks

#### A.3.0.3 Constants

The following constants are used within this model

$R_E$  the Earth's radius (km)

$$R_E = 6378.71$$

$c$  the speed of light in a vacuum (m/s)

$$c \simeq 3 \times 10^8$$

$k$  Boltzmann's constant (dBW/K/Hz)

$$k = -228.6$$

For all the following equations and formulas the  $P$  indicates a power, with subscripts G and L specifying gains and losses respectively. Whenever possible the description in parenthesis (description) indicates the source or associated device of the gain or loss.

#### A.3.1 Losses

##### A.3.1.1 Free Space Path Loss

For simplicity we will specify that the orbit is spherical. This implies that at a sun-synchronous orbit the following values can be used

$e$  the orbit's eccentricity ( $^\circ$ )

$$e \simeq 0.0 \tag{A.1}$$

$i$  the orbit's inclination ( $^\circ$ )

$$i \simeq 90 \tag{A.2}$$

Because the eccentricity is assumed to be  $0^\circ$  this implies that the Height of Perigee [hp] is equal to the Height of Apogee [ha]. For simplicity we will specify that the Height of Apogee is 700 kilometers.

$h_p$  the orbit's height of perigee (km)

$$h_p = 700 \tag{A.3}$$

For simplicity we will specify that the Height of Apogee is 700 kilometers.

From here we calculate the distances involved with the problem space.

$a$  the orbit's semi-major axis (km)

$$a = \frac{[ha + hp + 2(Re)]}{2} \quad (\text{A.4})$$

We arbitrarily assigned a value to  $\omega$  the Argument of Perigee ( $^\circ$ ) of

$$\omega \simeq 180 \quad (\text{A.5})$$

For similar reasons we arbitrarily assign a value to  $\Omega$  the Right Ascension of the Ascending Node (RAAN) ( $^\circ$ )

$$\Omega \simeq 180 \quad (\text{A.6})$$

Next we can determine  $\tau$  the orbital period (minutes)

$$\tau_{period} = 84.4892 [a/Re]^{1.5} \quad (\text{A.7})$$

Although not required for this model it may be instructive to know the rate of change of the Argument of Perigee and the rate of change of the RAAN respectively

$$\frac{d\omega}{dt} = 19.919482 \left(\frac{Re}{a}\right)^{3.5} \times \frac{1 - 1.25 \sin^2\left(\frac{I}{57.29578}\right)}{(1 - e^2)^2} \quad (\text{A.8})$$

$$\frac{d\Omega}{dt} = \left(\frac{-9.959704}{(1 - e^2)^2}\right) \left(\frac{Re}{a}\right)^{3.5} \cos\left(\frac{I}{57.29578}\right) \quad (\text{A.9})$$

The minimum  $\delta$  Elevation Angle ( $^\circ$ ) determines the possible Field of View (FOV) that the ground station is capable of communicating with satellites

$$\delta = 10 \quad (\text{A.10})$$

This value chosen as the minimum Elevation Angle due to the pointing capabilities of the ground station antenna and terrestrial obstructions.

Based on the Heights of Perigee and Apogee we can determine  $h$  the Mean Height Above the Surface (km)

$$h = \frac{ha + hp}{2} \quad (\text{A.11})$$



By using the relationship of the radius of the Earth and the Mean Height above the Surface

$$r = h + Re \quad (\text{A.12})$$

we can finally determine  $S$  the Slant Range (km) from the CubeSat to the ground station

$$S = Re \left( \sqrt{\frac{(h + Re)^2}{Re^2} - \cos^2 \delta} - \sin \delta \right) \quad (\text{A.13})$$

Next we need to know about  $\lambda$  the Wavelength (m) of the carrier frequency being used

$$\lambda = \frac{c}{f} \quad (\text{A.14})$$

Where  $f$  is the frequency used for the down link communications as seen described in Table A.2.

Finally we arrive at  $L_{path}$  the Simplified Free Space Path Loss in decibels (dB)

$$L_{path} = 22.0 + 20 \left( \log \frac{S}{\lambda} \right) \quad (\text{A.15})$$

### A.3.1.2 Atmospheric Loss

Clean air attenuation includes, rain, atmospheric signal scintillation, oxygen absorption, and suspended water vapor modeling. At microwave frequencies the phase angle of arrival can lead to approximately 2dB of loss. This is not applicable for our purpose but is included for completeness for future models.

Because of the wavelengths of interest according to Table A.2 we assume that the attenuation due to precipitation or  $L_{rain}$  Rain Loss (dB) is

$$L_{rain} \cong 0 \quad (\text{A.16})$$

According to the International Telecommunications Union (ITU)

Attenuation by atmospheric gases which are entirely caused by absorption depends mainly on frequency, elevation angle, altitude above sea level and water vapor density (absolute humidity). At frequencies below 10 GHz, it may normally be neglected [ITU2001].

Paffett *et al.* further support to that VHF is not impacted by atmosphere and ionosphere propagation [Paffett1998].

Realistically the atmospheric attenuation below 10 GHz is not zero but is between 5 and 6 orders of logarithmic magnitude lower than the free space path loss. For this model we can assume that  $L_{atmos}$  the Atmospheric Loss (dB) negligible due to the use of the 144 MHz frequency band. Detailed information for the atmospheric attenuation of frequencies above 1GHz are commonly available at via the ITU.

$$L_{atmos} \cong 0 \quad (\text{A.17})$$

### A.3.1.3 Ionospheric Loss

The ionospheric losses encountered by the transmission through the link path are based on the frequency used.

If we assume the height of the absorbing layer to be 100 km  $h_{iono}$  we can calculate the Ionospheric Loss  $L_{iono}(dB)$  as:

$$L_{iono}(f, \delta) = \frac{2200}{f^2 \sqrt{\sin(\delta)^2 + 2 \frac{h_{iono}}{R_E}}} \quad (\text{A.18})$$

Where  $R_E$  is the Earth's radius,  $f$  is the carrier frequency, and  $\delta$  is the elevation angle of the ground station antenna from the horizon [Boithias1987].

However this component was not used directly within this model. As an alternative we assumed a worst case scenario value from Table Table A.1 based on the frequency of choice. The Ionospheric Loss (dB) was set to

$$L_{iono} = 0.7 \quad (\text{A.19})$$

Frequency (MHz)	Attenuation (dB)
146	0.7
438	0.4
2410	0.1

**Table A.1:** Ionospheric Losses with frequency. After: [Thakker2010]

### A.3.1.4 Pointing Loss

Because the antenna may not be pointed in the correct location to line up with the ground station we determine  $L_{point}$  the Spacecraft Antenna Pointing Loss (dB) to be between 0 and 12. For simplicity we set this value to an idealized 0.

$$L_{point} = 0 \quad (\text{A.20})$$

Based on the pointing error specification of the YAESU g-5500 antenna rotor system used at UMARS as an example ground station we specify that  $\alpha_{point}$  the Ground Station Pointing Error ( $^{\circ}$ ) is

$$\alpha_{point} = \pm 4 \quad (\text{A.21})$$

Using the same UMARS system's  $\Theta$  the Ground Station Beamwidth ( $^{\circ}$ ) of the 2MCP22 antenna is

$$\Theta = 38 \quad (\text{A.22})$$

Similarly  $G_{ant}$  the Ground Station Peak Antenna Gain (dB) for the UMARS 2MCP22 antenna system at 145.9 MHz is

$$G_{ant} = 12.25 \quad (\text{A.23})$$

From the above values we can determine  $L_{gndpoint}$  the Ground Antenna Pointing Loss (dB)

$$L_{gndpoint} = -12 \left( \frac{\alpha_{point}}{\Theta} \right)^2 \quad (\text{A.24})$$

### A.3.1.5 Polarization loss

For the purposes of this simple model we will assume the spacecraft antenna is linearly polarized (vertical or horizontal with the direction of travel in the orbit) because its using an easy to build monopole antenna. Often these are made from precision cut lengths of tape measure.

The variation of ground stations by comparison will use a vertical whip antennas (monopoles) providing vertically polarized beams. For the model we could assume 3dB of Antenna Polarization Loss  $L_{polar}$  (dB)

$$L_{polar} = 3 \quad (\text{A.25})$$

However, this value will need to be statistically created based on model development.

### A.3.1.6 Multipath Loss

There's a fun one to think about....

I may have to consider all the stations as a mean distance above terrain. Then figure out what a typical quarter wave monopole has for multi-path at that height.

### A.3.2 Satellite Transmission system parameters

For the satellite downlink we must consider the following parameters

- the signal power
- the transmit frequency
- the Doppler frequency shift as seen by the receiving stations on the ground
- thermal noise
- transmitter intermodulation noise
- the Effective Isotropic Radiated Power from the satellite
- Spacecraft antenna radiation pattern
- spacecraft longitude
- slant range

Many CubeSats use existing hand held type amateur radios as transceivers. This design concept limits the output power of the transmitter to between 0 and 5 Watts. Within this model the user will specify a particular transmitter output power which is based on the available energy from the spacecraft power system and the transceiver's capabilities. The Spacecraft Transmit Power  $P_{tx}$  (W) is described by

$$0 \leq P_{tx}(W) \leq 5 \quad (\text{A.26})$$

In dBW that works out to

$$P_{tx}(dBW) = 10 (\log_{10} P_{tx}) \quad (\text{A.27})$$

In dBm that works out to

$$P_{tx}(dBm) = 10 (\log_{10} P_{tx}) + 30 \quad (\text{A.28})$$

The above dBm calculation is for comparative purposes only.

As a worst case scenario we will assume that the spacecraft's feed line to the antenna has significant length with poor coupling given a loss from 0 up to 3 dB along the length of the cable. This may be seen as an unrealistically high loss, providing a margin of error for the construction of the satellite link. For our first order approximation model we will assume Spacecraft Transmission Line Losses  $L_{line}$  of 0.5 dB

$$L_{line} \leq 0.5 \quad (\text{A.29})$$

We are assuming that the CubeSat antenna design itself will have either null net effect or have positive gain. It is possible to develop a high gain antenna for within the CubeSat specification. However highly directional antennas such as parabolic mesh dishes have difficulties in deployment within such a confined space. As such the Spacecraft Antenna Gain  $G_{ant}$  (dBiC) is

$$0 \leq G_{ant} \leq 18 \quad (\text{A.30})$$

As a worst case scenario we will assume that the spacecraft's feed line connector to the antenna has significantly poor coupling given a loss from 0 up to 3 dB to the cable. This may be seen as an unrealistically high loss, providing a margin of error for the construction of the satellite link. For simplicity of our first order approximation model we will assume the Spacecraft In-Line Connection Loss  $L_{connect}$  (dB) to be 0.5 dB

$$L_{connect} \leq 0.5 \quad (\text{A.31})$$

From Here we can determine the total output capability of the CubeSat transmission system, known as the spacecraft's Effective Isotropic Radiative Power (EIRP) (dBW)

$$EIRP = P_{tx} + L_{line} + G_{ant} + L_{connect} \quad (\text{A.32})$$

Once all of the above is determined we can then calculate the Isotropic Signal Level at Ground Station  $P_{rx}$  (dB)

$$P_{rx} = EIRP + L_{point} + L_{polar} + L_{polar} + L_{iono} + L_{rain} \quad (\text{A.33})$$

### A.3.3 Ground Receiver system parameters

As a worst case scenario we will assume that the ground station feed line to the antenna has significant length with poor coupling given a loss of 3 dB along the length of the cable and its connector. This may be seen as an unrealistically high loss, providing a margin of error for the construction of the satellite link. The Ground Station Transmission Line Loss  $L_{gndline}$  (dB) was set to

$$L_{gndline} = 3 \quad (\text{A.34})$$

The typical ground station Low-Noise-Amplifier (LNA) Noise Temperature ( $^{\circ}K$ ) was used

$$T_{LNA} = 125 \quad (\text{A.35})$$

LNA are used by ground stations to amplify the weak signals received from spacecraft. According to [Minoli2009] an LNA is a low-noise device that receives and amplifies satellite

signals at the output of the antenna feed horn, without changing the input frequency, and instilling minimal amounts of noise to the received signal. For commercial applications in the 1GHz and above frequencies LNA's provide between 50-60dB of gain [Minoli2009].

By similar conventions the Ground Station transmission Line Temperature ( $^{\circ}K$ ) was set to 290

$$T_{txline} = 290 \quad (\text{A.36})$$

And the Ground Station Sky Temperature  $T_{gndsky}$  ( $^{\circ}K$ ) was set to

$$T_{gndsky} = 450 \quad (\text{A.37})$$

The  $C_{gndline}$  value describes the ability for EM wave propagation across the coupling from the antenna to the feed line, as well as from the feed-line to the transceiver. The Ground Station Transmission Line Coefficient (dB) is

$$C_{gndline} = 10^{\frac{L_{gndline}}{10}} \quad (\text{A.38})$$

From here we can determine the effects of noise sources in the ground station receiver system.

Ground Station Effective Noise Temperature  $T_{gndnoise}$  ( $^{\circ}K$ )

$$T_{gndnoise} = C_{gndline} + T_{gndsky} + (1 - C_{gndline})T_{txline} + T_{LNA} \quad (\text{A.39})$$

This provides a step towards the Ground Station Figure of Merit  $(G/T)(dB/^{\circ}K)$

$$FOM_{gnd} = G_{ant} + L_{gndline} - 10\log_{10}(T_{gndnoise}) \quad (\text{A.40})$$

The Ground Station LNA Noise Temperature ( $^{\circ}K$ ) was set to

$$T_{LNA} = 125 \quad (\text{A.41})$$

Finally we can determine the effective energy received. The Ground Station SNR Power Density  $SNR_{gnd}$  (dBHz) is

$$SNR_{gnd} = P_{rx} + L_{point} - k + G_{ant} \quad (\text{A.42})$$

### A.3.4 Criteria of the Digital Reception

$E_b/N_0$  is defined as the ratio of Energy Per Bit ( $E_b$ ) to the Noise Power Spectral Density ( $N_0$ ) which is used as a measure of how “strong” a signal is within a digital transmission system. If  $E_b/N_0 < 0$  it implies that the signal and the noise power levels are equal. Typically satellite communications systems use a digital modulation scheme such as Phase-Shift Keying, Frequency-Shift Keying, Amplitude-Shift Keying or Quadrature Amplitude Modulation in order to convey more information upon the carrier signal. These different forms of modulation each have different theoretical Bit Error Rates ( $BER$ ) vs  $E_b/N_0$ . As such we will limit our discussion for simplicity to using Shannon-Hartley theorem limit calculations.

Range	Band (m)	Min Freq. (MHz)	Max Freq. (MHz)	Channel width (MHz)
VHF	2	144	146	0.02
UHF	0.7	435	438	0.025
UHF	0.23	1260	1270	0.025 from 1294.025 to 1294.175
UHF	0.23	1260	1270	0.100 from 1299.05 to 1299.95
UHF	0.13	2400	2410	Assumed 10 MHz
UHF	0.13	2430	2438	Assumed 8 MHz
SHF	0.09	3400	3410	Assumed 10 MHz
SHF	0.05	5650	5670	Assumed 20 MHz
SHF	0.05	5830	5850	Assumed 20 MHz

**Table A.2:** Channelization of AMSAT frequency bands. After:[Shtun1995]

First we select a frequency band of interest from Table A.2, which determines a channel bandwidth available to us based on Table A.2 under normal conditions. For this first level approximation model we specify that our channel bandwidth will be 0.02 MHz increments according to the channel band plans outlined by the Radio Amateurs of Canada [Shtun1995] as shown in Table A.2. The ITU Amateur-Satellite Service is outlined by [ITU2012a].

The text in red indicates uplinks only and are included for completeness.

This is a large generalization as for example, the 2 m band has distinct channelization requirements whereas the 0.05 m band has none. This allows us to attempt to compare the available frequency bands with each other. Further analysis should be done to demonstrate how the variation in channel bandwidth impacts the calculations.





## B Appendix: Surveys

**Table B.1:** Survey of I-Gate use

Do you operate an APRS I-Gate	Count	%
I use it continuously (24/7)	6	37.5
I use it several times a week	1	6.3
I use it a few times a week	3	18.8
I use it a few times a year. (special events, etc.)	0	0
I use it once, possibly twice a year or less	0	0
Its been more than a year since I've used it	1	6.3
Someone else has borrowed it from me	0	0
I don't recall the last time I used it	0	0
I've never actually used it	5	31.3

**Table B.2:** Survey of I-Gate use 2

If you operate an APRS I-Gate, how often is it in use?	Count	%
Continuously (24/7)	9	81.8
Weekdays	0	0
Weekends	0	0
Special occasions (ham fests, field days, other events)	1	9.1
Occasionally	1	9.1
Rarely	0	0

**Table B.3:** Survey of AMSAT/OSCAR ground station use

Do you operate a satellite ground station?	Count	%
Yes	7	43.8
No	9	56.3

Do you own an antenna like the one shown above?

Response	Count	Percent
Yes	12	66.7
No	6	33.3

**Table B.4:** Survey of AMSAT/OSCAR ground station antennas

If you do operate a satellite ground station, can you describe its antenna?	Count	%
J-Pole type	0	0
Whip type	1	14.3
Yagi-Uda type	6	45.7
4 element ground plane with a vertical whip type	0	0
Other	2	28.6

**Table B.5:** Survey of I-Gate antenna types

If you do operate an APRS I-Gate, please tell me what design of antenna do you use for it?	Count	%
J-Pole type	2	18.2
Whip type	7	63.6
Yagi-Uda type	1	9.1
4 element ground plane with a vertical whip type	2	18.2
"Eggbeater" type	0	0
Other	1	9.1

If you own an antenna like the one show, how often do you use it?

Response	Count	Percent
I use it continuously (24/7)	3	27.3
I use it several times a week	2	18.2
I use it a few times a week	0	0
I use it a few times a year. (special events, etc.)	3	27.3
I use it once, possibly twice a year or less	0	0
Its been more than a year since I've used it	3	27.3
Someone else has borrowed it from me	0	0
I've never actually used it	0	0
I don't recall the last time I used it	0	0

Did you buy or build an antenna similar to the one shown above?

**Table B.6:** Survey of AMSAT/OSCAR ground station use

If you operate an APRS I-Gate, how often is it in use?	Count	%
Continuously (24/7)	1	14.3
Weekdays	1	14.3
Weekends	1	14.3
Special occasions (ham fests, field days, other events)	1	14.3
Occasionally	3	42.9
Rarely	1	14

Response	Count	Percent
I bought it fully assembled	4	36.4
I built it "from scratch"	5	45.5
I borrowed it	1	9.1
I bought it as a kit that had to be built	1	9.1
It was a gift that came as a kit that had to be built	0	0

Have you seen the above antenna used for an APRS I-Gate?

Response	Count	Percent
Yes, I've seen it used for an APRS I-Gate	7	58.3
No, I haven't seen it used for an APRS I-Gate	5	41.7

If you use an antenna similar to the one shown, on a scale from 1 to 5, how would you rate its performance overall?

Response	Count	Percent
Very Good: 1	0	0
2	6	54.5
3	4	36.4
4	1	9.1
Very Bad: 5	0	0

If you use an antenna similar to the one shown, on a scale from 1 to 5, how would you rate its performance during inclement radio frequency noise?

Response	Count	Percent
Very Good: 1	0	0
2	5	45.5
3	5	45.5
4	1	9.1
Very Bad: 5	0	0

If you use an antenna similar to the one shown, on a scale from 1 to 5, how would you rate its performance during inclement weather?

Response	Count	Percent
Very Good: 1	1	9.1
2	3	27.3
3	5	45.5
4	1	9.1
Very Bad: 5	1	9.1

Was an antenna like the one shown above used for APRS (at your station or someone else's)?

Response	Count	Percent
Yes	7	58.3
No	4	33.3
I'm not sure	1	8.3

Have you seen an antenna like the one shown above used for an APRS I-Gate (at your station or someone else's)?

Response	Count	Percent
Yes	5	41.7
No	5	41.7
I'm not sure	2	16.7

On a scale of 1 to 5, how difficult would it be to build the antenna design shown above?

Response	Count	Percent
Very Good: 1	0	0
2	0	0
3	4	28.6
4	4	28.6
Very Bad: 5	6	42.9

How common is the antenna design shown above?

Response	Count	Percent
Very Common	8	53.3
Common	7	46.7
Not Sure	0	0
Rare	0	0
Very Rare	0	0

A small and informal survey was conducted via social media to determine what transmit power operators typically used for their mobile APRS beacons:

For your mobile APRS, what is your beacon output power from the transmitter to the antenna?

**Table B.7:** Community survey of mobile APRS beacon transmit power

For your mobile APRS, what is your beacon output power from the transmitter to the antenna?

Response	Count	Percent
Less than 1 Watt	6	6%
Above 1 Watt but less than 5 Watts	47	45%
Above 5 Watts but less than 10 Watts	21	20
Above 10 Watts but less than 35 Watts	15	14
35 Watts or more	15	14

A small and informal survey was conducted via social media to determine what beacon interval operators typically used for their mobile APRS beacons:

**Table B.8:** Community survey of mobile APRS beacon interval

For your mobile APRS, what is your beacon interval set to? In other words; If your first beacon is at time = 0 how much time will elapse before your next beacon?

This question is assuming we're not talking about smart beaconing based on speed, distance traveled or heading changes.

Response	Count	Percent
30 s	5	9
60 s (1 min)	16	30
90 s (1 min & 30 s)	4	8
120 s (from 2 min to under 3 min)	7	13
180 s (3min or more)	21	40



# C Appendix: Source Code

This appendix contains samples of some of the source code used within this thesis. Note that the alignment of the code has been modified to display it as text.

Special thanks to Dario Schor, Matt Woelk and Rob Barwell for their help by creating this code.

## C.1 APRS I-Gate Call Signs and Locations

### C.1.1 record

The first step in collecting the I-Gate locations is to record a live feed of packets streaming from the Tier 2 servers. This is step 1 of the process and records the parts of the live feed we want. At the top of the file one can change the prefixes of interest. One may also see that at about midway through the code is a line for restricting the latitude/longitude to restrict the area of interest.

```
#      Name      record.pl
#
#      Description
#      This is step 1 in the process and used to
#      record qAR packets from the live stream.
#
#      Input Params
#      Param 1 – Filename to store the iGate call sign to
```

```
use Ham::APRS::IS;
use Ham::APRS::FAP qw(parseaprs);

# A list of valid call sign prefixes.
# More can be added with the | sign and ^ sign
$valid="^4A|^4B|^4C|^4V|^6D|^6E|^6F|^6G|^6H|^6I
|^6J|^6Y|^AA|^AB|^AC|^AD|^AE|^AF|^AG
|^AH|^AI|^AJ|^AK|^C6|^CF|^CG|^CH|^CI|^CJ|^CK
|^CL|^CM|^CO|^CY|^CZ|^FG|^FJ|^FM
|^FP|^FS|^HH|^HI|^J7|^K|^KP1|^KP2|^KP3|^KP4|^N
|^NP1|^NP2|^NP3|^NP4|^OX|^PJ5|^PJ5
|^PJ6|^PJ7|^T4|^TD|^TG|^V2|^V3|^V4|^VA|^VB|^VC
|^VD|^VE|^VF|^VG|^VO|^VP2E|^VP2M
|^VP2V|^VP5|^VP9|^VX|^VY|^W|^WP1|^WP2|^WP3|^WP4
```

```

|^XA|^XB|^XC|^XD|^XE|^XF|^XG|^XH
|^XI|^XJ|^XK|^XL|^XM|^XN|^XO|^ZF";
# Stop time is current time in seconds plus some offset.
# i.e. for 3 hours change to time+(60*60*3)
$stopTime=time+(60*80);

if($ARGV[0] eq "")
{
    print "Please input file name, i.e.
    \"record.pl myfilename.txt\"";
} else
{
    open(RECFILE, '>' . $ARGV[0]);
    # login to the APRS IS master stream server.
    #Note the filter parameters for qAR packets and
    # area from 80N -170W to 6N -44W
    $conn = new Ham::APRS::IS('rotate.aprs.net:14580',
    'REPLACE-WITH-YOUR-CALLSIGN', 'appid' =>
    'REPLACE-WITH-YOUR-APP-NAME', 'filter' =>
    'q/rR a/80/-170/6/-44');
    $conn->connect('retryuntil' => 3) || die
    "Failed to connect: $is->{error}";

    print "Start Record: " . localtime(time) . "\n";
    while(time <= $stopTime)
    {
        $packet = $conn->getline_noncomment();
        if(defined $packet)
        {
            $packet = uc($packet);
            # ensures it is a qAR packet
            if($packet =~ m/QAR, (.*)/)
            {
                $station = $1;
                # make sure it has a valid prefix
                # based on the settings above in $valid
                if($station =~ m/$valid/)
                {
                    print RECFILE $station . "\n";
                    print "found=" . $station . "\n";
                }
            }
        }
    }
    print "End Record: " . localtime(time) . "\n";
}

```



```

    $conn->disconnect() || die
    "Failed to disconnect: $is->{error}";
    close(RECFILE);
}

```

List of North American amateur radio call signs prefixes by country

- Bermuda (UK) VP9
- Canada XK XM VX VD XO CZ CG VA CH VF CJ CK VE XJ VC VY XL CF XN  
VB VG CI CY VO
- Greenland (Den.) OX
- Mexico XD XH XC XF XA XE XG XB XI 4A 4C 4B 6G 6E 6H 6J 6F 6D 6I
- Saint Pierre and Miquelon (Fr.) FP
- United States AA AH AF AD K AE AC AJ N W AG AI AB AK
- Anguilla (UK) VP2E
- Antigua and Barbuda V2
- Bahamas, The C6
- British Virgin Islands (UK) VP2V
- Cayman Islands (UK) ZF
- Cuba CM T4 CL CO
- Dominica J7
- Dominican Republic HI
- Guadeloupe (Fr.) FG
- Haiti HH 4V
- Jamaica 6Y
- Martinique (Fr.) FM
- Montserrat (UK) VP2M
- Navassa Island (US) KP1 WP1 NP1
- Puerto Rico (US) KP4 WP3 KP3 NP3 NP4 WP4
- Saba (Neth.) PJ5 PJ6

- Saint Barthélemy (Fr.) FJ
- Saint Kitts and Nevis V4
- Saint Martin (Fr.) FJ FS
- Sint Eustatius (Neth.) PJ5
- Sint Maarten (Neth.) PJ7
- Turks and Caicos Islands (UK) VP5
- United States Virgin Islands (US) KP2 NP2 WP2
- Guatemala TG TD
- Belize V3

### C.1.2 sort

This is step two of the process where it takes the list and distills it down to unique call sign

```
#      Name      sort.pl
#
#      Description
#      This is step 2 in the process and
#      used to remove the duplicate call signs.
#
#      Input Params
#      Param 1 – Filename from record.pl with
#      the list of station call signs
#      Param 2 – Filename to save
#      the unique call sign list to

if ($ARGV[0] eq "")
{
    print "Please input file name, i.e.
    \"sort.pl infilename.txt outfilename.txt\"";
} else
{
    %allStations={};
    open(SORTFILE,'<'.$ARGV[0]);
    while(<SORTFILE>)
    {
        $line=$_;
        $line=~s/\n|\r//g;
        $allStations{$line}=$allStations{$line}+1;
    }
}
```

```

        close(SORTFILE);

        open(OUTFILE, '>>'.$ARGV[1]);
        foreach $key (keys %allStations)
        {
            print OUTFILE $key."\n";
        }
        close(OUTFILE);
    }
}

```

### C.1.3 getinfo

This is the last step of the process where it produces the final file by doing look ups on APRS.fi for the lat and long of each call sign.

```

#      Name      getinfo.pl
#
#      Description
#      This is step 3 in the process and used to
#      get the lat and long of the call signs.
#
#      Input Params
#      Param 1 – Filename from sort.pl with the list
#                 #of unique station call signs
#      Param 2 – Filename to save the tab seperated list
#                 #of call sign , lat , long

use LWP::UserAgent;
use HTTP::Request::Common;
use JSON;

#      the maximum number of stations per query to APRS.fi
$maxStations=19;

#      the time to sleep in seconds between getting data
$timePause=60*5;

if ($ARGV[0] eq "")
{
    print "Please input file name, i.e.
    \"getinfo.pl infilename.txt outfilename.txt\"";
} else
{
    open(INFILE, "<".$ARGV[0]);
    open(OUTFILE, ">>".$ARGV[1]);

```

```

$currCount=0;
$currReq="";
while(<INFILE>)
{
    $line=$_;
    $line=~s/\n|\r//g;
    if ($currCount<$maxStations)
    {
        $currReq=$line." ";
        $currCount++;
    } elseif ($currCount==$maxStations)
    {
        $currReq=$line;
        $currCount++;
    } else
    {
        &getInfo($currReq);
        sleep($timePause);
        $currReq=$line." ";
        $currCount=1;
    }
}
close(OUTFILE);
close(INFILE);
}

#      Name      getInfo
#
#      Description
#      Submits a query for each group of
#      maxStations for lat and long
#
#      Input Params
#      Param 1 – List of stations to query

sub getInfo
{
    $query=$_[0];
    # URL for APRS.fi
    $url="http://api.aprs.fi/api/get?name=".$query.
        "&what=loc&apikey=
        REPLACE-WITH-YOUR-API-KEY&format=json ";
    $ua = LWP::UserAgent->new;
    $req = POST $url;
    $res = $ua->request($req);

```

```

    if ($res->is_success)
    {
        $json_res=$res->decoded_content;
        $stationInfo= decode_json($json_res);
        foreach my $station
        ( @{ $stationInfo->
            {entries} } )
        {
            print OUTFILE $station->
                {name}."\t". $station->{lat} .
                "\t". $station->{lng}."\n";
        }
    } else
    {
        print $res->status_line . "\n";
    }
}

```

### C.1.4 R statistical programming sample

This is a short sample of the type of R code used to create the White and Brown Noise data series and then test it by collecting different moments, autocorrelation values or PP tests. Much of the code is commented out as rather than generate new files for each set of required data the code was simply modified to generate it on demand.

```

# Requires:
# 1) tseries
# 2) RobPer

# Close all plots
graphics.off()

set.seed(123)

# Define constants for simulation
numPoints      <- 23308
windowSizeList <- c(50,100,150,200,250,300,350,400,450,500,
,550,600,650,700,750)
windowSize     <- 0
alphaBrown     <- 2

# Array to store the stationarity data
# for each window size
# and each window.

```

```
acfDataBrown <- 0

# Array to store points to plot the moments
firstMomentBrown <- 0
secondMomentBrown <- 0

# Generate data set in the time domain and
# normalize it between 0 and 1.
brown = TK95(numPoints, alphaBrown)
brown = (brown + abs(min(brown))) /
        (max(brown) - min(brown))

# Iterate through every window size
currWindowSizeIndex <- 1
while(currWindowSizeIndex <= length(windowSizeList))
{
  # Pick the current window size from the list
  windowSize <- windowSizeList[currWindowSizeIndex]

  # Calculate how many windows fit based
  # on the length of the data.
  # Note the windows must be full, so we use the
  # floor function
  # to round number.
  numPointsInWindows <- floor(length(brown) /
    windowSize) * windowSize

  # Define an array to store all the data
  acfLength = length(acf(brown[indexStart:indexEnd],
    type=c("correlation"), plot=FALSE)$acf)
  acfDataBrown <- matrix(0, nrow=floor(length(brown)/
    windowSize), ncol=4+acfLength)

  # Iterate through every window in the data
  currIndex <- 1
  currWindowIndex <- 1
  while(currIndex <= numPointsInWindows)
  {
    # Index of start and end of window
    indexStart <- currIndex
    indexEnd <- currIndex + windowSize - 1

    # Calculate moments
    # The moment arrays store windowSize data points
    # of the same value
```

```
# for plotting purposes the same way it was
# performed in my thesis.

#firstMomentBrown[indexStart:indexEnd]
#      <- mean(brown[indexStart:indexEnd])
#secondMomentBrown[indexStart:indexEnd]
#      <- var(brown[indexStart:indexEnd])

# Calculate Phillip–Perron test: Null hypothesis
#pp <- suppressWarnings(pp.test(brown[indexStart:
#      indexEnd], alternative = c("stationary")))

# Calculate Autocorrelation test:
acf <- acf(brown[indexStart:indexEnd],
           type = c("correlation"), plot=FALSE)

# Computes the Kwiatkowski–Phillips–Schmidt–Shin
# (KPSS) test for the null
# hypothesis that x is level or trend stationary.
#kpssLevel <- suppressWarnings
#      (kpss.test(brown[indexStart:indexEnd],
#      null = c("Level")))
#kpssTrend <- suppressWarnings(kpss.test(brown
#      [indexStart:indexEnd], null = c("Trend")))

# Column 1: Window Number
columnName = c("Window Number")
acfDataBrown[currWindowIndex,1] <- currWindowIndex

# Column 2: Mean
columnName = c(columnName, "Mean")
acfDataBrown[currWindowIndex,2] <- mean(brown
      [indexStart:indexEnd])

# Column 3: Variance
columnName = c(columnName, "Variance")
acfDataBrown[currWindowIndex,3] <- var(brown
      [indexStart:indexEnd])

# Column 4: Covariance
columnName = c(columnName, "Covariance")
acfDataBrown[currWindowIndex,4] <- cov(seq(1,
      windowSize), brown[indexStart:indexEnd])

# Column 5: autocorrelation of autocovariance data
```

```
acfIndex <- 1
while(acfIndex <= acfLength)
{
  columnName = c(columnName, paste(" acf values",
                                     acfIndex))
  acfDataBrown[currWindowIndex,5:5+acfIndex-1]
    <- acf$acf[acfIndex]
  acfIndex <- acfIndex + 1
}

# Column 6: pp.test p-value
#columnName = c(columnName, "pp.test p-value")
#acfDataBrown[currWindowIndex,6] <- pp$p.value

# Column 7: pp.test parameter
#columnName = c(columnName, "pp.test parameter")
#stationarityData[currWindowIndex,7] <- pp$parameter

# Column 8: pp.test method
#columnName = c(columnName, "pp.test method")
#stationarityData[currWindowIndex,8] <- pp$method

# Column 9: KPSS Level statistic
#columnName = c(columnName, "KPSS Level statistic")
#stationarityData[currWindowIndex,9]
#      <- kpssLevel$statistic

# Column 10: KPSS Level p-value
#columnName = c(columnName, "KPSS Level p-value")
#stationarityData[currWindowIndex,10]
#      <- kpssLevel$p.value

# Column 11: KPSS Level method
#columnName = c(columnName, "KPSS Level method")
#stationarityData[currWindowIndex,11]
#      <- kpssLevel$method

# Column 12: KPSS Trend statistic
#columnName = c(columnName, "KPSS Trend statistic")
#stationarityData[currWindowIndex,12]
#      <- kpssTrend$statistic

# Column 13: KPSS Trend p-value
#columnName = c(columnName, "KPSS Trend p-value")
#stationarityData[currWindowIndex,13]
```



```

        #           <- kpssTrend$p.value

# Column 14: KPSS Trend method
#columnName = c(columnName, "KPSS Trend method")
#stationarityData[currWindowIndex,14]
        #           <- kpssTrend$method

# Advance to the next window.
currIndex          <- currIndex + windowSize
currWindowIndex <- currWindowIndex + 1
}

# Save to a CSV
filename <- paste("BrownWindowACFcollection",
windowSize, ".csv", sep="")
write.table(acfDataBrown, file=filename,
sep="," , col.names=columnName, row.names=FALSE)

# Advance to the next window size
currWindowSizeIndex <- currWindowSizeIndex + 1
}

```

## C.2 Tools Used

Throughout the development of this thesis a number of different tools were used. What follows is a brief list with a description of how and why there were chosen. They are listed in no particular order.

**Lyx:** A document process that acts as a front end to  $\text{\LaTeX}$  which itself is a typography system. Lyx allowed me to use  $\text{\LaTeX}$  and focus on the content rather than staring at the document's source code all day. Its a what you see is what you mean system. For example you can't use two spaces in a row. Layout is determined by the document class file which is similar to CSS for html pages. Its output is a high quality PDF for publication.

**$\text{\LaTeX}$ :** A typesetting system that includes features designed for creating technical and scientific documents.  $\text{\LaTeX}$  does very well at larger documents. It keeps track of formulas, section headings and everything else. If one plans to do math with formula numbers,  $\text{\LaTeX}$  is very likely in your future.

**MS\_Excel:** The ubiquitous MS Excel was very useful for rapid data exploration. So far its also the fastest way to create useful and detailed graphs.

**MS\_PowerPoint:** The ubiquitous presentation software. Used because it is everywhere and it works.

**MS\_Visio:** A proprietary diagramming tool used to create a few of the graphics for the thesis.

**Dia:** A free and open source diagramming tool used to create a few of the graphics for the thesis.

**Google\_Sheets:** In a manner similar to MS Excel Google Sheets was useful for data exploration. More importantly it allowed me to quickly collect data, reformat it and then share it.

**Google\_Drive:** Everything in this thesis was saved to a Google Drive. This allowed me to share documents and data between my own computers as well as anyone else who needed it.

**Dropbox:** Initially I had used this online data storage service. I migrated to Google Drive when my employer blacklisted the Dropbox domain on their firewall for very valid security reasons.

**Google\_Forms:** The amateur radio community survey was created and deployed using Google Forms. It was free, easy to use and I didn't have to purchase a separate web domain or hosting.

**Google\_Chrome:** My browser of choice. At this point we can see that Google has access to much of my data/brain.

**Google\_Slides:** An online presentation tool. Less rich in features than MS PowerPoint but has the distinct advantage of online collaboration. This tool was used by UMSATS to create a presentation by approximately 17 students simultaneously. This prevented version control and merging issues. The time advantage was immense.

**Google\_Diagrams:** An online tool for creating many of the SVG files within the thesis.

**STK:** This was the primary software used for the discrete event simulation. This was discussed in Sec. 3.2. I was already familiar with it and it could do both the orbital mechanics and the communications models accurately.

**4nec2:** An antenna modelling software often used by the amateur radio community. This was discussed in Sec. 3.2. It was straight forward to use and it's also free.

**GNU\_Radio\_Companion:** This is a Linux software tool for the development and experimentation of Software Defined Radios. This was discussed in Sec. 3.4. It's free and has a graphical user interface. The domains of expertise to use it are daunting. One must understand Digital Signal Processing, Radio hardware and spectrums, and the idiosyncrasies of being on the edge of Linux software development. This was discussed in Sec. 3.4.

**HackRF:** A software defined radio available from Great Scott Gadgets. This was discussed in Sec. 3.4. The HackRF was chosen because it was known to interface with GNU Radio Companion and was less expensive than the other leading options.

**MS\_Windows\_OS:** The operating system of choice because STK is not available for any other. Nor is 4nec2. During the writing of the thesis my desktop was forced to move from Windows 8 to Windows 10. Its been ....interesting.

**MS\_Surface\_Pro\_(1):** This small form factor laptop/tablet was used for any truly portable computing needs. Initially it was capable of displaying STK but driver/OS updates now prevent it. As an added benefit the stylus input allows for note taking of math and other diagrams which was very useful.

**SDR#:** This is a Software Defined Radio program that can interface with the HackRF. This was chosen for its ubiquity to understand how the HackRF functioned before moving on to GNU Radio Companion.

**R\_Studio:** An integrated development environment for the R statistical language. This was discussed in Chapter 5. It was chosen because of its robust features, better graphical user interface for the neophyte and its free. It also saves to SVG.

**Freeplane:** This mind mapping tool was used during the initial stages of the thesis to create document outlines and track sections. Its free and open source.

**Foxit\_Reader:** A free, but proprietary PDF viewer and editor. This was used to view the compiled outputs from Lyx. It also allows you to fill out form field and add digital signatures. This is handy for a part time student who may not have time to cross town and fill in a form.

**Inkscape:** A vector graphics software tool that is free and open source. This was used to convert graphics to SVG format. For example it was possible to save graphs from MS Excel as a PDF, import them into Inkscape, modify them and then save them as SVG files to be imported into Lyx. Convolved but effective.

**Geogebra:** Another free and open source tool used for math visualizations and creating graphs. It includes features and capabilities not possible with MS Excel or not practical in R Studio. It also saves to SVG.

**JabRef:** A free and open source Bib<sub>T</sub>E<sub>X</sub> database management tool. With this I was able to collate all of the references and easily import them into Lyx.

**Gimp:** A free and open source image editing and creation software tool. This was used to create some graphics and convert between formats.

**Greenshot:** A free and open source screen capture tool. This was used to collect images from STK.

**7-zip:** An open source software tool for file compression.

**Raspberry\_PI\_model\_B:** Three of these low cost computers on a single PCB were used as small form factor I-Gates as described in Sec.3.4.

**SDR:** Three NooElec R820T SDR & DVB-T NESDR Mini was used and the software defined radio, along with the three Raspberry-Pi. This was chosen for its cost, popularity and shipping in Canada.

**Kenwood\_D700:** This mobile radio was used to test packet address fields as described in Sec. 2.4.3. It was chosen because it was already installed in the vehicle. was installed in a truck and set to have no path information for its outgoing packets.

**Yeasu\_FTM400\_Delta:** This mobile radio was used to test packet address fields as described in Sec. 2.4.3. It was chosen because it was already installed in the vehicle. was installed in a truck and set to have no path information for its outgoing packets.

# D Appendix: Supplementary Analysis Data, Results and Methods

**Table D.1:** Access Summary Report from ExampleSat-165 to VE4UM I-Gate

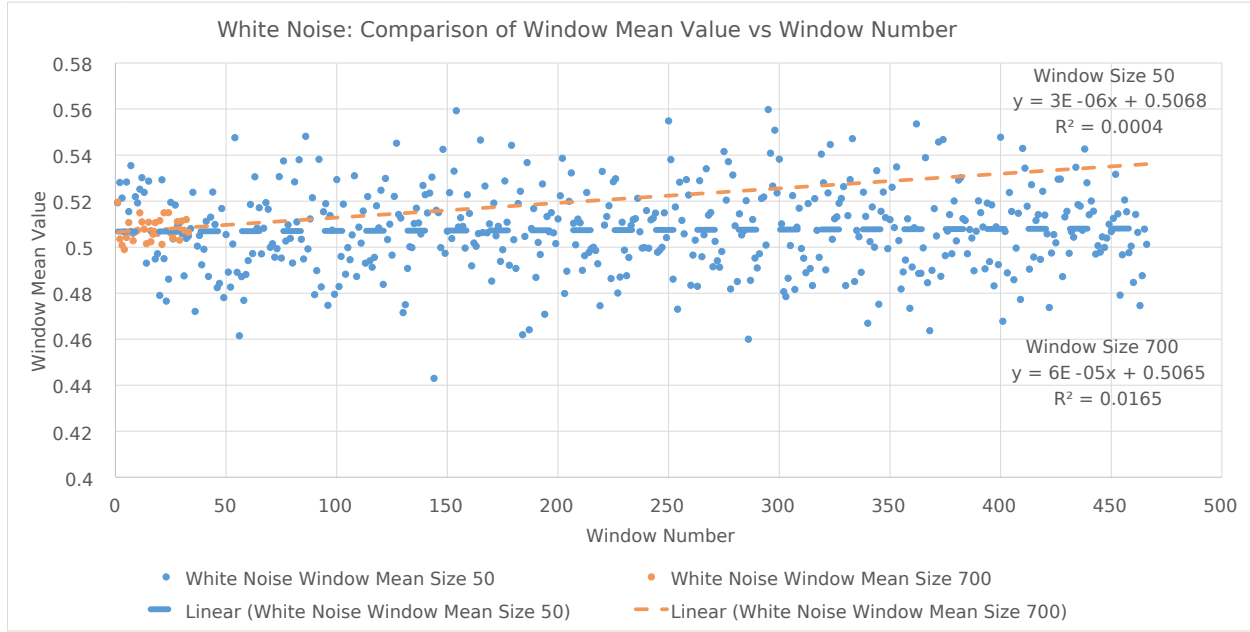
Access	Start Time	Stop Time	Duration (s)
1	21 Oct 2015 05:30:23.308	21 Oct 2015 05:33:38.127	194.849
2	21 Oct 2015 06:56:12.712	21 Oct 2015 06:56:50.077	37.5
...	...	...	...
10	21 Oct 2015 18:24:52.687	21 Oct 2015 18:25:44.884	52.197

Min Duration	36.302
Max Duration	280.406
Mean Duration	113.072
Total Duration	1130.716

**Table D.2:** Link Budget Report from ExampleSat-165 to VE4UM I-Gate (1 s interval)

Time	EIRP (dBW)	Rcvd. Freq. (Hz)	Rcvd. Iso. Pwr (dBW)	Flux density (dBW/m <sup>2</sup> )
21 Oct 2015 05:30:23.308	1.633	144391830.164464	-181.355	176.708080
21 Oct 2015 05:30:24.000	1.658	144391820.744011	-175.769	171.122573
...	...	...	...	...
21 Oct 2015 16:53:45.100	7.571	144393195.287505	-132.565	-127.919022

Time	g/T (dB/K)	C/N0 (dB * Hz)	Bandwidth (kHz)	C/N (dB)	E <sub>b</sub> /N <sub>0</sub> (dB)	BER
21 Oct 2015 05:30:23.308	-23.119253	24.145379	7.000	-14.3256	-6.6664	3.212625e-001
21 Oct 2015 05:30:24.000	-22.119306	30.710832	7.000	-7.7401	-0.0810	1.609111e-001
...	...	...	...	...	...	...
21 Oct 2015 16:53:45.100	11.002110	107.035800	7.000	68.5848	76.2440	1.000000e-030



**Fig. D.1:** Plot of White Noise short time window means vs window number with comparative slopes; window sizes 50 and 700 samples

## D.1 Time Series Analysis

### D.1.1 Short Time Window Analysis of Non-Stationary Time Series

#### D.1.1.1 Short Time Window Mean plots

#### D.1.2 Shapiro-Wilks

A detailed derivation of the Shapiro-Wilks test is beyond the scope of this thesis.

$$W = \frac{\left(\sum_{i=1}^n a_i x_{(i)}\right)^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (\text{D.1})$$

Where:

$x_{(i)}$  is the  $i$ th order statistic and thus is the smallest value within the sample set.

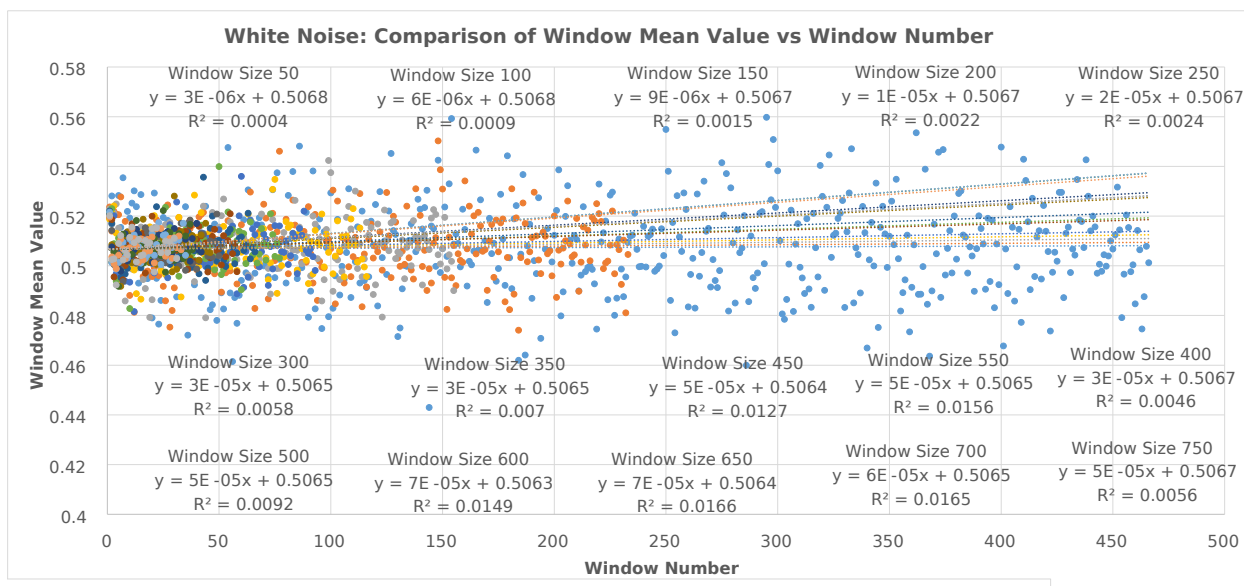
$\bar{x} = (x_1 + \dots + x_n) / n$  is the mean of the sample set.

$a_i$  is determined by:

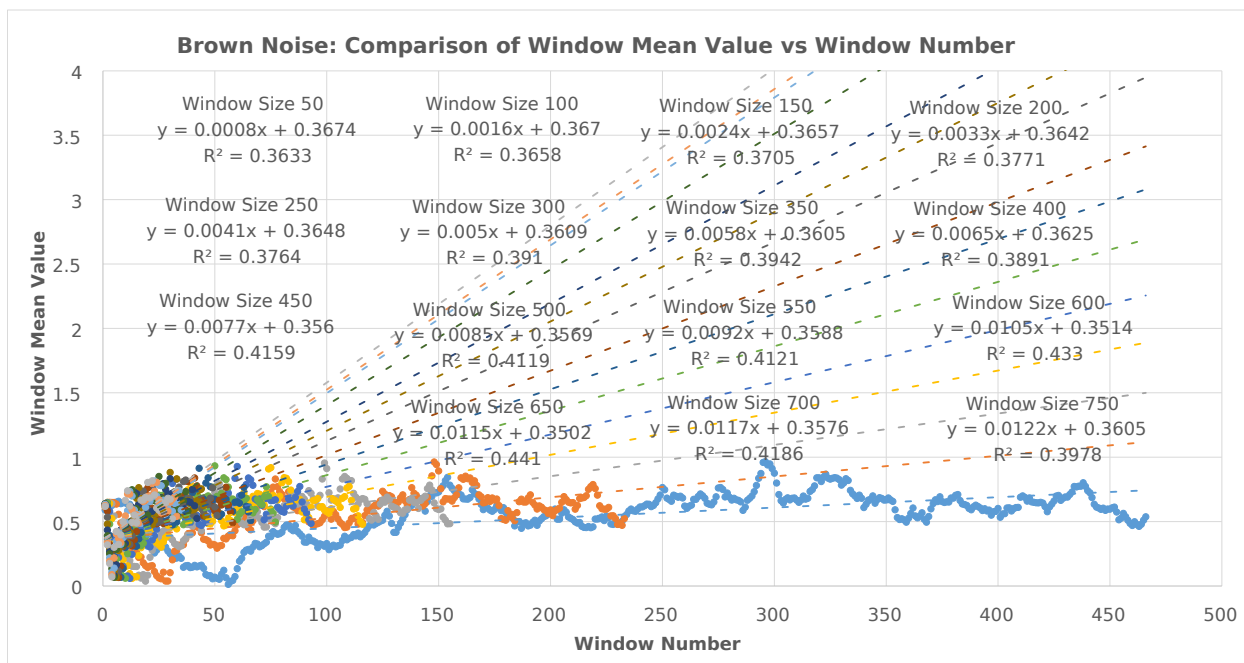
$$(a_1 + \dots + a_n) = \frac{m^\top V^{-1}}{(m^\top V^{-1} V^{-1} m)^{1/2}}$$

Given that

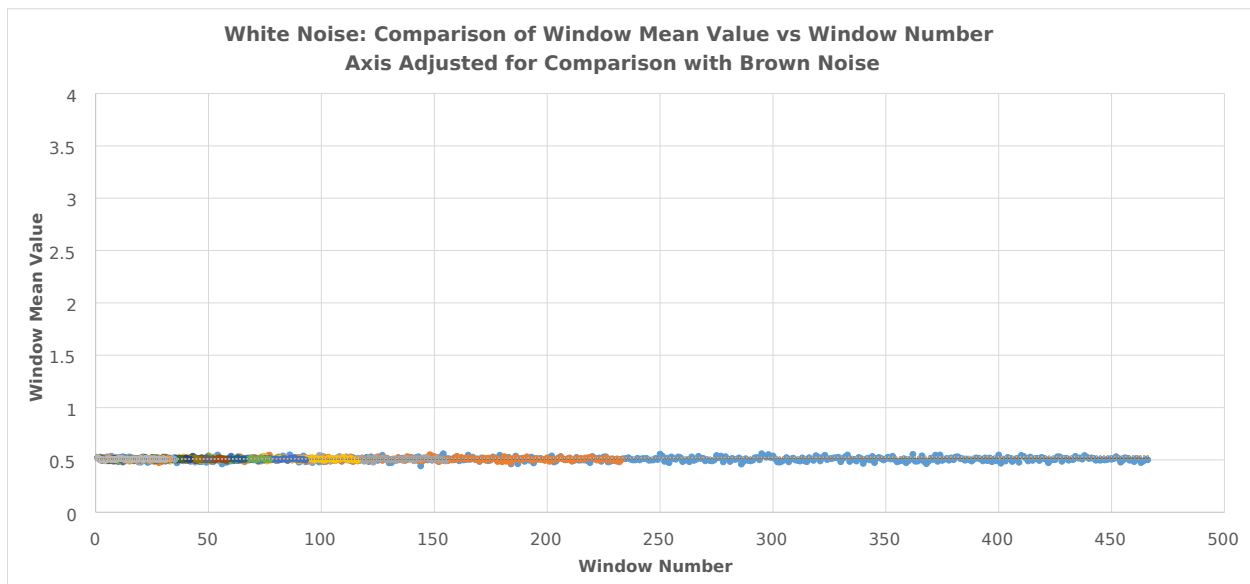
$m = (m_1, \dots, m_n)^\top$  and  $m_1, \dots, m_n$  are the expected values of the order statistics, of the assumed to be independent and identically distributed, random variable which was sample from the standard normal distribution.  $V$  is the covariance matrix of those order statistics.



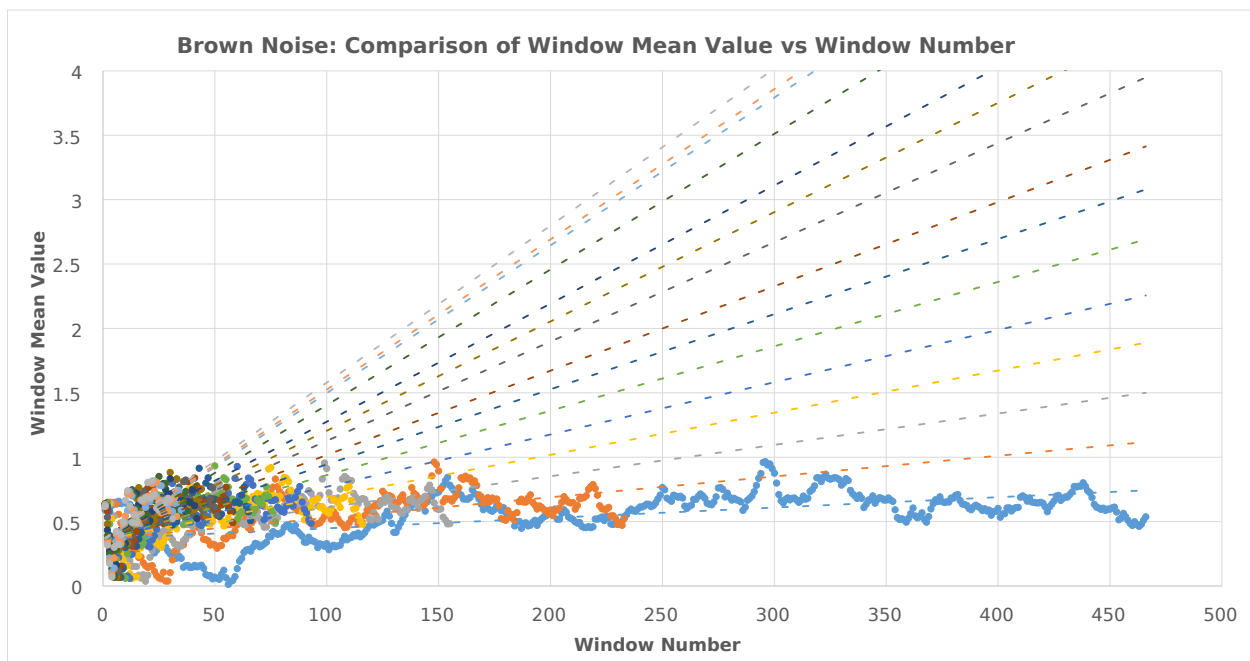
**Fig. D.2:** Plot of White Noise short time window means vs window number with comparative slopes



**Fig. D.3:** Plot of Brown Noise short time window means vs window number with comparative slopes

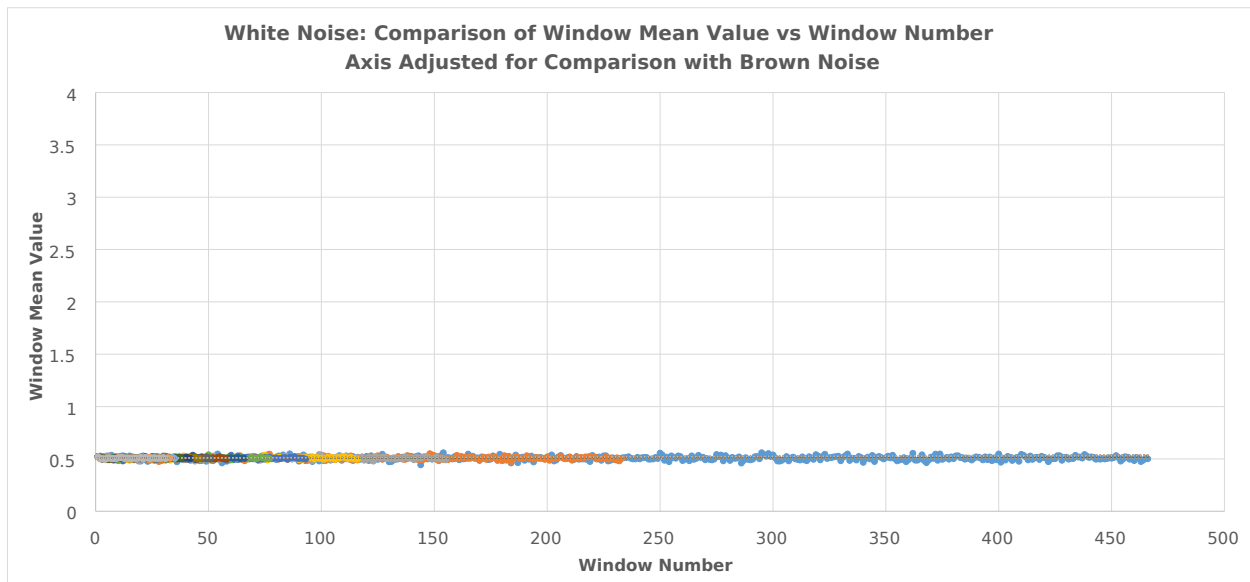


**Fig. D.4:** Plot of White Noise short time window means vs window number, axis adjusted for comparisons

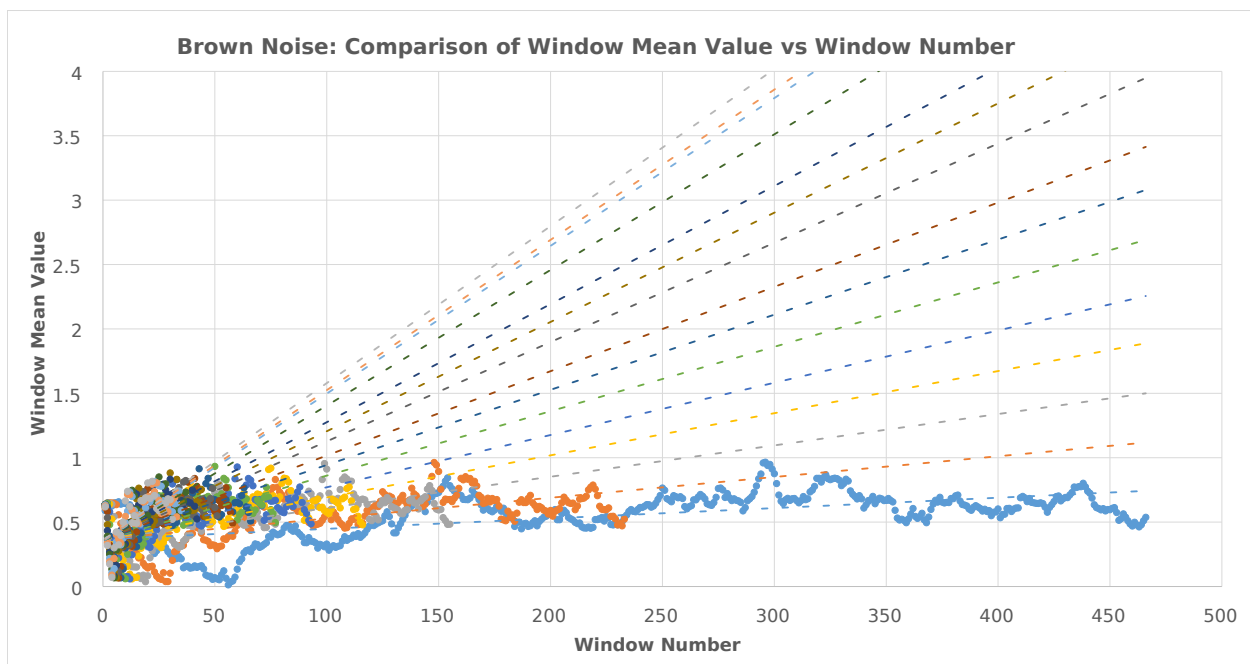


**Fig. D.5:** Plot of Brown Noise short time window means vs window number, display adjusted for comparisons



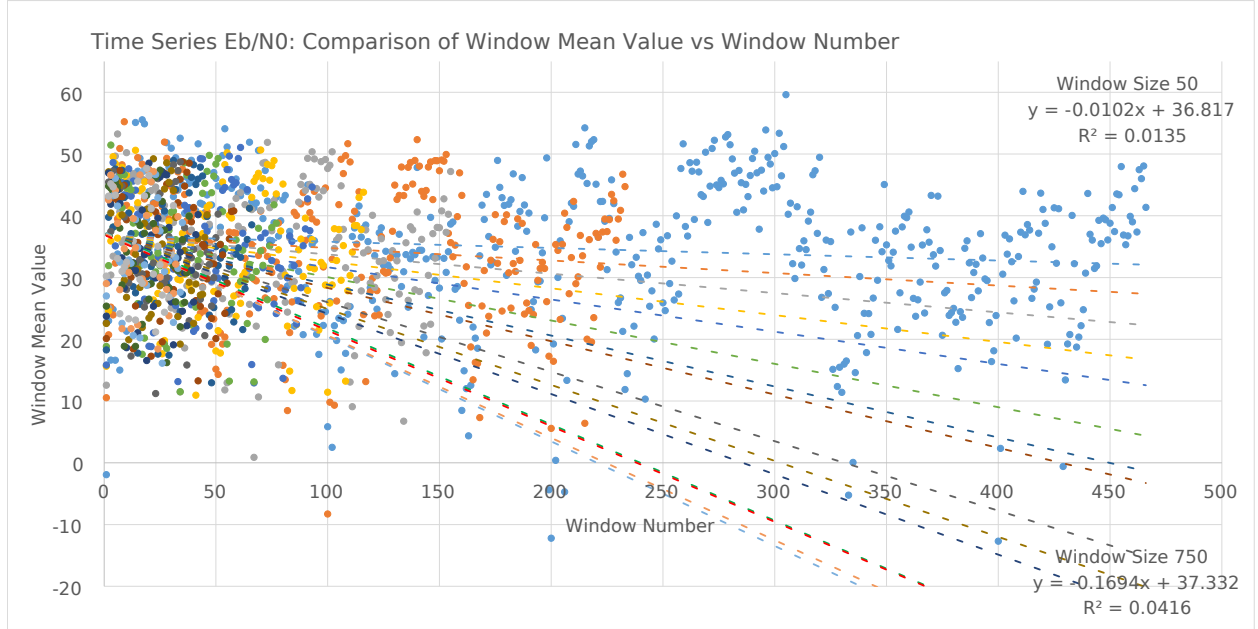


(a) Plot of White Noise short time window means vs window number, axis adjusted for comparisons



(b) Brown Noise

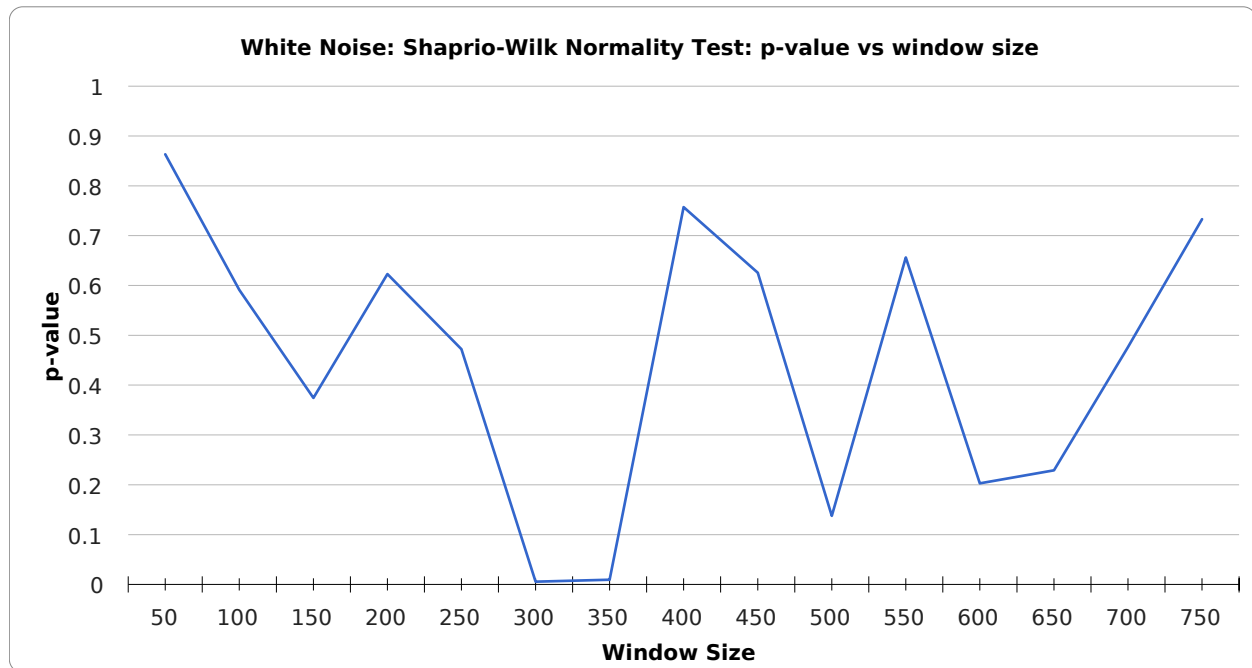
**Fig. D.6:** Plot of Brown Noise short time window means vs window number, display adjusted for comparisons



**Fig. D.7:** Plot of  $E_b/N_0$  Window mean value vs window number with comparative slopes for all short time window sizes

**Table D.3:** White Noise Shapiro-Wilk Normality Test

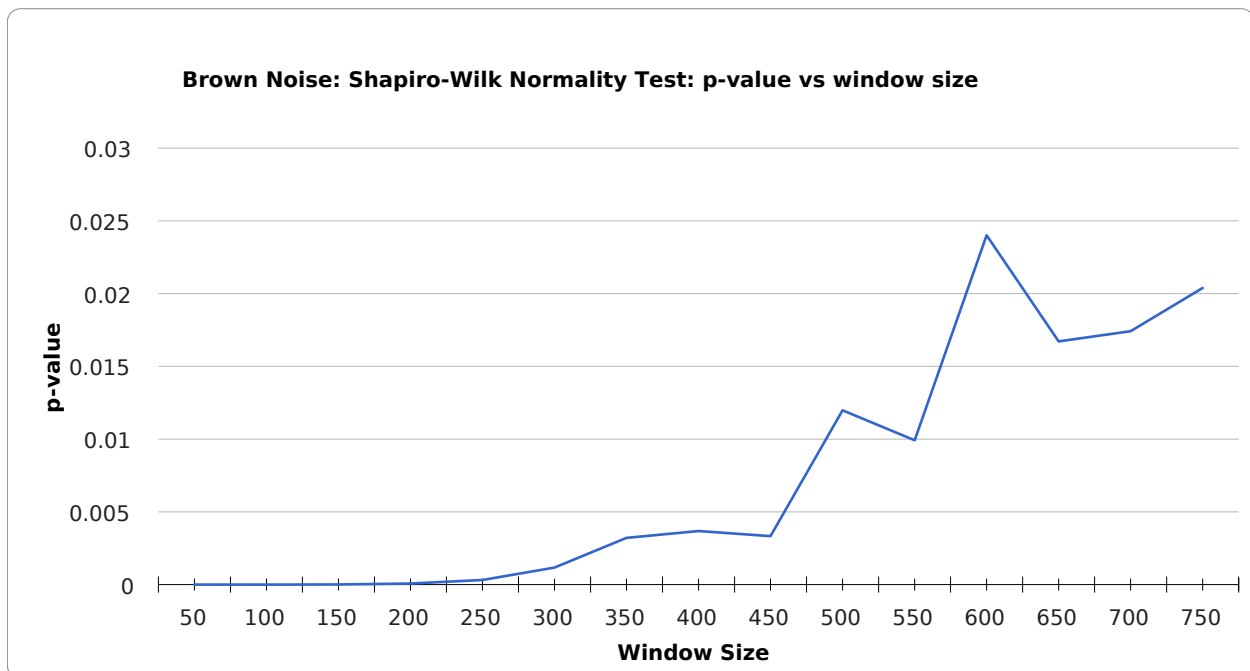
Window Size	W	p-value
50	0.99799	0.8633
100	0.99469	0.5912
150	0.99038	0.3742
200	0.99071	0.6231
250	0.98672	0.4721
300	0.95193	0.005517
350	0.94978	0.009576
400	0.98633	0.7572
450	0.98198	0.6256
500	0.96204	0.1377
550	0.97987	0.6563
600	0.9609	0.203
650	0.96002	0.229
700	0.96986	0.4763
750	0.97727	0.7332



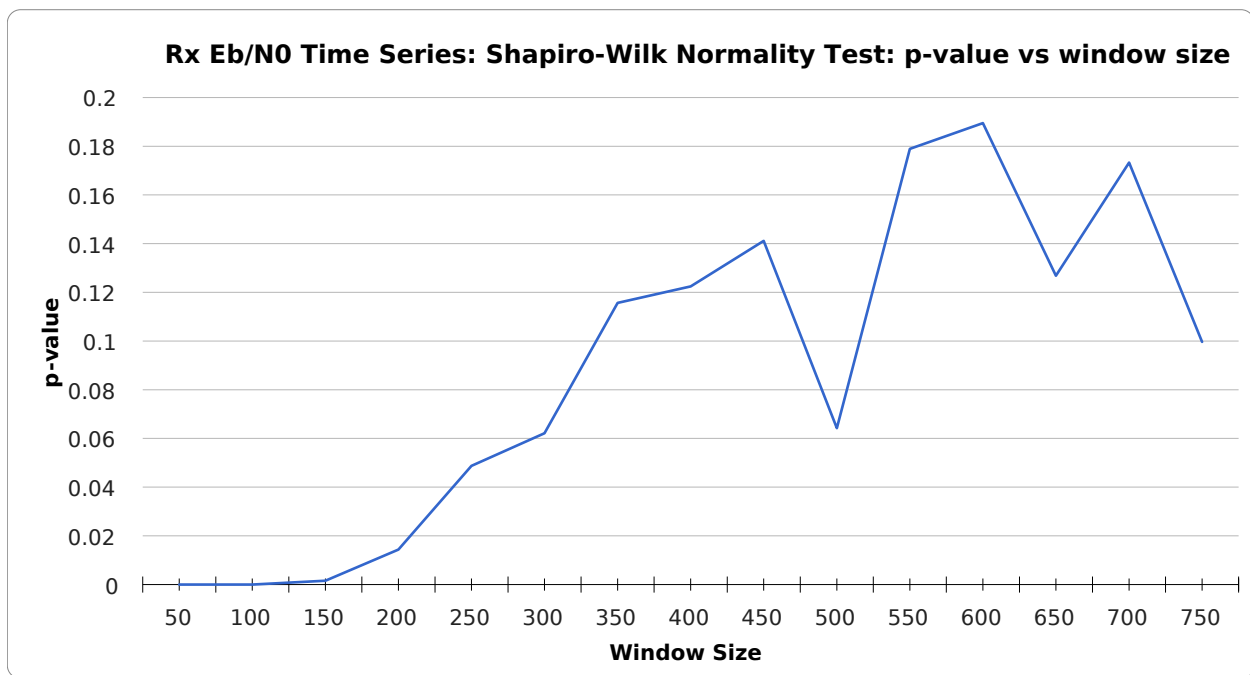
**Fig. D.8:** White Noise Shapiro-Wilk Normality Test p-value vs Window Size

**Table D.4:** Brown Noise Shapiro-Wilk Normality Test

Window Size	W	p-value
50	0.94873	1.26E-11
100	0.94693	1.64E-07
150	0.94521	9.74E-06
200	0.94204	7.85E-05
250	0.93947	0.0003111
300	0.93954	0.001171
350	0.94008	0.003212
400	0.9343	0.003682
450	0.92541	0.003335
500	0.93422	0.01198
550	0.92655	0.009924
600	0.93247	0.02401
650	0.92238	0.01671
700	0.91921	0.01742
750	0.9176	0.02039

**Fig. D.9:** Brown Noise Shapiro-Wilk Normality Test p-value vs Window Size**Table D.5:** Rx  $E_b/N_0$ Shapiro-Wilk Normality Test

Window Size	W	p-value
50	0.95631	1.60E-10
100	0.96478	1.62E-05
150	0.96933	0.001575
200	0.97156	0.01434
250	0.97276	0.04872
300	0.96964	0.06214
350	0.9704	0.1157
400	0.96754	0.1224
450	0.96536	0.1411
500	0.95353	0.0642
550	0.96233	0.1789
600	0.95998	0.1895
650	0.95159	0.1268
700	0.95395	0.1733
750	0.94296	0.09965



**Fig. D.10:**  $E_b/N_0$  Time Series Shapiro-Wilk Normality Test p-value vs Window Size